ΠΑΝΕΠΙΣΤΗΜΙΟ ΠΑΤΡΩΝ
ΠΟΛΥΤΕΧΝΙΚΗ ΣΧΟΛΗ
ΤΜΗΜΑ ΜΗΧΑΝΙΚΩΝ Η/Υ
ΚΑΙ ΠΛΗΡΟΦΟΡΙΚΗΣ

Διπλωματική Εργασία

ΜΕΛΕΤΗ ΤΕΧΝΟΛΟΓΙΑΣ MOBILE IPv6 ΜΕ ΧΡΗΣΗ ΕΞΟΜΟΙΩΤΗ

Υπεύθυνος Καθηγητής: Αναπληρωτής Καθηγητής Χρήστος Μπούρας
Επιβλέπων: κ. Κώστας Στάμος

Αριστομενόπουλος Γιώργος ΑΜ: 3009

Πάτρα, Σεπτέμβριος 2007

ΠΕΡΙΕΧΟΜΕΝΑ

ΚΑΤΑΛΟΓΟΣ ΕΙΚΟΝΩΝ

ΑΚΡΩΝΥΜΙΑ

3G	3rd Generation
3GPP	3rd Generation Partnership Project
ANG	Access Network Gateways
AP	Access Point
ARP	Address Resolution Protocol
BA	Binding Acknowledgments
BCMP	BRAIN Candidate Mobility Management Protocol
BS	Base Station
BU	Binding Update
CDMA	Code Division Multiple Access
CDS	Conceptual Data Structures
CIP	Cellular IP
CN	Correspondent Node
CoA	Care of Address
CoT	Care-of-address Test
CoTI	Care-of-address Test Init
DAD	Duplicate Address Detection
DHCP	Dynamic Host Configuration Protocol
DHCPv6	Dynamic Host Configuration Protocol Version 6
DRL	Default Routers List
EBA	Early Binding Acknowledgement
EBU	Early Binding Update
EMA	Edge Mobility Architecture

FA	Foreign Agent
FBack	Fast Binding Acknowledgement
FBU	Fast Binding Update
F-HMIPv6	Fast Hierarchical MIPv6
FNA	Fast Neighbor Advertisement
GTP	GPRS Tunneling Protocol
HA	Home Agent
HACK	Handover Acknowledgment
HAWAII	Handoff-Aware Wireless Access Internet Infrastructure
HI	Handover Initiate
HMIPv6	Hierarchical Mobile IPv6
HoA	Home Address
HoT	Home-address Test
HoTI	Home-address Test Init
ICMP	Internet Control Message Protocol
IDMP	Intra-domain Mobility Management Protocol
IEEE	Institute of Electrical and Electronic Engineers
IETF	Internet Engineering Task Force
IMS	Internet Multimedia Service
IP	Internet Protocol
IPng	Next Generation Internet Protocol
IPv4	Internet Protocol Version 4
IPv6	Internet Protocol Version 6
ISO	International Organization for Standardization
L2	Layer 2 of OSI model, or Data Link layer

L3	Layer 3 of OSI model, or Network layer
LAN	Local Area Network
LBU	Local Binding Update
LCoA	Local Care-of-Address
LMA	Localized Mobility Agent
LMM	Localized Mobility Management
MAC	Media Access Control
MAP	Mobility Anchor Point
MCMT	Mobile Controlled Movement Tracking
MIPv4	Mobile IPv4
MIPv6	Mobile IPv6
MN	Mobile Node
NA	Neighbor Advertisement
NAT	Network Address Translation
NAR	New Access Router
NCoA	New Care of Address
NDP	Neighbor Discovery Protocol
NETLMM	Network-Based Localized Mobility Management
NS	Neighbor Solicitation
NUD	Neighbor Unreachability Detection
ODAD	Optimistic Duplicate Address Detection
OSI	Open System Interconnection reference model
PAR	Previous Access Router
PCoA	Previous Care of Address
PCoAF	Previous Care of Address Forwarding

PrRtAdv	Proxy Router Advertisement
RBU	Regional Binding Update
RCoA	Regional Care-of-Address
RCF	Request For Comments
RtSolPr	Router Solicitation for Proxy Advertisement
RA	Router Advertisement
RS	Router Solicitation
RTT	Round Trip Time
SHTR	Strong Handoff Radio Trigger
SMR	Session-to-mobility ratio
TCP	Transmission Control Protocol
TORA	Temporally Ordered Routing Algorithm
TLA	Top Level Aggregation
TTL	Time To Live
UMTS	Universal Mobile Telecommunications System
VoIP	Voice over IP
WLAN	Wireless LAN

Πρόλογος

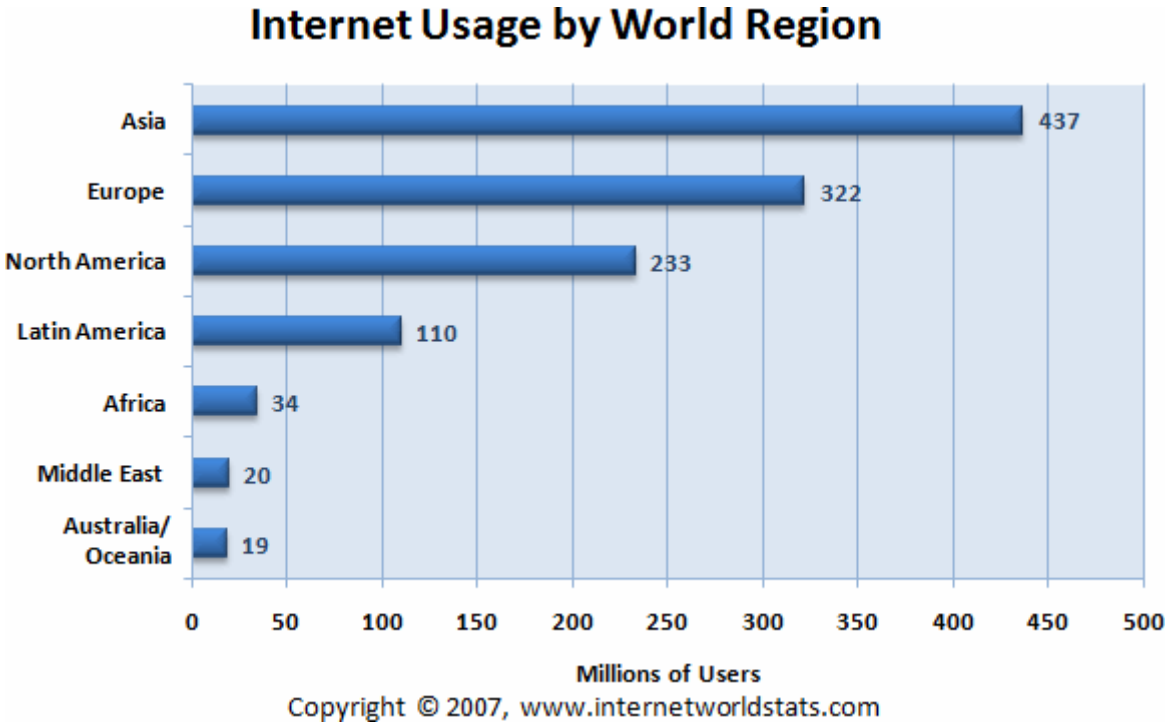
Στόχος της διπλωματικής είναι η παρουσίαση και μελέτη του πρωτοκόλλου Mobile IPv6, καθώς και διαφόρων προτεινόμενων επεκτάσεων του, που εγγυώνται καλύτερους handover χρόνους και συνεπώς μικρότερη απώλεια δεδομένων κατά την μετακίνηση μεταξύ διαφορετικών δικτύων.

Η διπλωματική αυτή χωρίζεται σε δύο μέρη. Το πρώτο μέρος περιλαμβάνει τα πρώτα 4 κεφάλαια και αποτελεί μια αναλυτική παρουσίαση των εννοιών IPv6 και Mobile IPv6, καθώς και άλλων εννοιών που τις συνοδεύουν. Το δεύτερο μέρος αποτελείται από τα κεφάλαια 5 έως 7 και αναφέρεται στην πειραματική ανάλυση και αξιολόγηση διαφόρων τεχνικών που εγγυούνται μικρότερη handover καθυστέρηση.

Πιο συγκεκριμένα το 1^{ο} κεφάλαιο παρουσιάζει την τρέχουσα αλλά και την μελλοντική κατάσταση στον χώρο των τηλεπικοινωνιών και την αναγκαιότητα εισαγωγής του πρωτοκόλλου IPv6 στις τηλεπικοινωνίες. Το 2^{ο} κεφάλαιο περιγράφει την έννοια των handovers, αναλύει την δομή του πρωτοκόλλου IPv6 και Mobile IPv6, όπως επίσης και την διαδικασία μετάβασης από το δημοφιλές IPv4 στο IPv6. Στο επόμενο κεφάλαιο παρουσιάζονται διάφορες τεχνικές και βελτιώσεις του MIPv6 πρωτοκόλλου που έχουν προταθεί κατά καιρούς με στόχο την ελαχιστοποίηση της handover καθυστέρησης. Στο 4^{ο} κεφάλαιο σχολιάζεται και αναλύεται η έννοια της ιεραρχικής δομής ενός MIPv6 δικτύου και τα πλεονεκτήματα που ενδεχομένως εισάγει η χρήση της. Στο πρώτο κεφάλαιο του δεύτερου μέρους, γίνεται η παρουσίαση του εξομοιωτή OMNet++ και του μοντέλου πάνω στο οποίο θα εργαστούμε για την αξιολόγηση των τεχνικών βελτίωσης της handover καθυστέρησης. Στο κεφάλαιο 6, παραθέτουμε και σχολιάζουμε τα αποτελέσματα των μετρήσεων μας καταλήγοντας στην καλύτερη κατά τη γνώμη μου τεχνική. Τέλος στο 7^{ο} κεφάλαιο γράφουμε συνοπτικά τα συμπεράσματα μας από αυτήν την διπλωματική.

1. Εισαγωγη

Η ανάπτυξη του Διαδικτύου έχει επιφέρει τεράστιες αλλαγές στον κόσμο των υπολογιστών και των επικοινωνιών. Από τα πρώτα παγκόσμια δίκτυα πληροφοριών έως την κυριαρχία του Παγκόσμιου Ιστού, από το ARPANET έως το MP3, οι υπολογιστές έχουν αλλάξει τον τρόπο που ο κόσμος αλληλεπιδρά. Στατιστικά στοιχεία [1] δείχνουν πως σχεδόν 1,2 δισεκατομμύρια άνθρωποι χρησιμοποιούν αυτή τη στιγμή το Internet, αριθμός που αντιπροσωπεύει το 18% του παγκόσμιου πληθυσμού. Ιδιαίτερη αύξηση στον αριθμό των χρηστών παρατηρείται σε αναπτυσσόμενες περιοχές, όπως η Μέση Ανατολή, η Αφρική και η Λατινική Αμερική, με δείκτες ανάπτυξης που ξεπερνούν το 500% σε σχέση με το 2006.



Εικόνα 1: Χρήση Internet ανά γεωγραφική περιοχή

Εκτός όμως από την αύξηση χρήσης του Internet, αυξάνονται και οι απαιτήσεις των χρηστών. Η ανάγκη για μόνιμη σύνδεση στο Διαδίκτυο, ακόμα και στον δρόμο, γίνεται όλο και πιο επιτακτική. Σύμφωνα με την NOKIA, μέχρι το 2009 θα υπάρχουν πάνω από 3 δισεκατομμύρια χρήστες κινητών τηλεφώνων παγκοσμίως με δυνατότητες πρόσβασης στο Internet, ενώ σύμφωνα με την Probe Group, θα υπάρχουν περίπου 600 εκατομμύρια χρήστες ασύρματου Internet μέχρι το 2008. Ο Alan Mosher, Διευθυντής Ερευνών της Probe Group, δηλώνει πως «Καθώς οι κινητές συσκευές γίνονται όλο και πιο φτηνές, οι διακομιστές θα γίνουν πιο επιθετικοί στο marketing και την τιμολογιακή τους πολιτική».

Ασύρματες τηλεπικοινωνίες θα γίνουν διαθέσιμες σε περιοχές όπου πριν δεν υπήρχε καν τηλέφωνο. Ενδεικτικό παράδειγμα [2] η Ινδία όπου μόλις το 17% έχει τηλέφωνο. Τα περισσότερα από τα νεώτερα δίκτυα που χτίζονται, ακόμη και εκείνα στις αναπτυσσόμενες αγορές, θα είναι πιθανώς 2.5G EDGE ή υψηλότερα, επιτρέποντας στους κατοίκους να χρησιμοποιούν τον Παγκόσμιο Ιστό. Άνθρωποι που πότε δεν συνδέθηκαν στο Διαδίκτυο από έναν υπολογιστή γραφείου, θα αποκτήσουν ξαφνικά ασύρματη πρόσβαση. Ο τρόπος με τον οποίο οι άνθρωποι θα δουλεύουν και ζουν θα αλλάξει. Η μόνιμη πρόσβαση στο Internet από οπουδήποτε θα θεωρείται πια δεδομένη.

Οι σημερινές τηλεπικοινωνίες αποτελούνται από ένα συνοθύλευμα ετερογενών δικτύων, συνδεδεμένων μεταξύ τους με περίπλοκες τεχνικές. Το πρωτόκολλο IP έχει επιλεχτεί σαν μέσο σύγκλισης αυτών των δικτύων. Το πρωτόκολλο IP είναι ήδη δοκιμασμένο και τυποποιημένο από την Internet Engineering Task Force (IETF), ενώ παράλληλα η αναμφίβολη κυριαρχία του σαν δίκτυο υποδομής, καθιστούν αναμφίβολα σοφή αυτή την επιλογή. Στα κινητά δίκτυα τρίτης γενιάς 3G ήδη το IP χρησιμοποιείται σαν πρωτόκολλο υποδομής, επιτρέποντας σε ετερογενείς συσκευές να αποκτούν πρόσβαση στο Internet.

Τα σημερινά κινητά δίκτυα χρησιμοποιούν την 4^{η} έκδοση του πρωτοκόλλου IP [3]. Το Mobile IPv4 επιτρέπει σε κινητούς κόμβους να μένουν μόνιμα συνδεδεμένοι στο Internet, ανεξάρτητα από τη θέση τους στο δίκτυο, διατηρώντας την ίδια IP διεύθυνση. Ακριβέστερα σκοπός του Mobile IPv4 είναι να κάνει την κίνηση του

κινητού κόμβου αόρατη προς τις εφαρμογές και τα πρωτόκολλα υψηλότερου επίπεδου, όπως το TCP.

Η ραγδαία ανάπτυξη όμως των χρηστών Internet δημιουργεί προβλήματα στη περαιτέρω χρήση του IPv4. Προβλήματα όπως ο περιορισμένος χώρος διευθύνσεων και η ¨ακαταστασία¨ του δικτύου θα λυθούν με την έλευση της έκδοσης 6 (IPv6) του πρωτοκόλλου IP. Ο θεωρητικός χώρος διευθύνσεων για το IPv6 ανέρχεται περίπου στις 340×10^{36} διευθύνσεις, συγκριτικά με τις $4{,}3 \times 10^{9}$ του IPv4, δημιουργώντας έτσι ένα σχεδόν άπειρο εύρος διαθέσιμων διευθύνσεων. Αυτό επιλύει προφανώς το πρόβλημα έλλειψης IP διευθύνσεων, αλλά επιλύει και το πρόβλημα ¨ακαταστασίας¨ που μπορεί να επιφέρει μεγάλο overhead καθώς το δίκτυο ολοένα και αυξάνεται.

Παρότι το MIPv6 φαντάζει ιδανικό στην αντιμετώπιση των παραπάνω προβλημάτων, παρότι υπάρχουν ήδη πολλές πλατφόρμες που το υποστηρίζουν και παρότι υπόσχεται αυξημένη ασφάλεια, δεν είναι ένα τελειοποιημένο πρωτόκολλο έτοιμο για χρήση, αλλά παραμένει ένα έργο σε εξέλιξη από την IETF. Ένα από τα βασικά προβλήματα που αντιμετωπίζει και με το οποίο θα ασχοληθούμε είναι τα λεγόμενα handovers ή handoffs. Ο πλέον διαδεδομένος τρόπος για να συμβεί ένα handover είναι η μετάβαση του κινητού χρήστη (MN) από την περιοχή κάλυψης μίας κυψέλης στην περιοχή κάλυψης μιας γειτονικής κυψέλης. Κατά τη μετάβαση αυτή ο MN χάνει την σύνδεση με την κυψέλη του και έως ότου συνδεθεί με την γειτονική δεν είναι ικανός να στείλει ή να λάβει οποιαδήποτε πληροφορία. Το χρονικό αυτό διάστημα είναι πολύ κρίσιμο, καθώς όσο πιο μεγάλο είναι τόσο πιο πολλά πακέτα θα χάνει. Έχουν προταθεί διάφορες τεχνικές στα πλαίσια της τυποποίησης του MIPv6 που υπόσχονται να μειώσουν στο ελάχιστο τον χρόνο μετάβασης και συνεπώς και τον αριθμό των χαμένων πακέτων. Σκοπός αυτής της μελέτης είναι η εκτίμηση των προτεινόμενων τεχνικών μέσω εξομοίωσης και η εύρεση ενός κατώτατου ορίου για αυτό το διάστημα.

Στο επόμενο κεφάλαιο δίνουμε μια πιο αναλυτική περιγραφή για τα handover σε IP δίκτυα, την μετάβαση από το IPv4 στο IPv6, καθώς και τα βασικά χαρακτηριστικά την τεχνολογίας Mobile IPv6. Στο τρίτο και τέταρτο κεφάλαιο περιγράφουμε τις προτεινόμενες τεχνικές που επιδιώκουν καλύτερους χρόνους μετάβασης. Το πέμπτο

κεφάλαιο περιγράφει το περιβάλλον εξομοίωσης που θα χρησιμοποιήσουμε, όπως επίσης και τα διάφορα σενάρια για την αξιολόγηση της κάθε τεχνικής. Στο κεφάλαιο 6 παραθέτουμε και σχολιάζουμε τα αποτελέσματα που βρήκαμε, ενώ τελικά στο έβδομο κεφάλαιο ανακεφαλαιώνουμε και κλείνουμε αυτή την έρευνα με προτάσεις για περαιτέρω πειράματα.

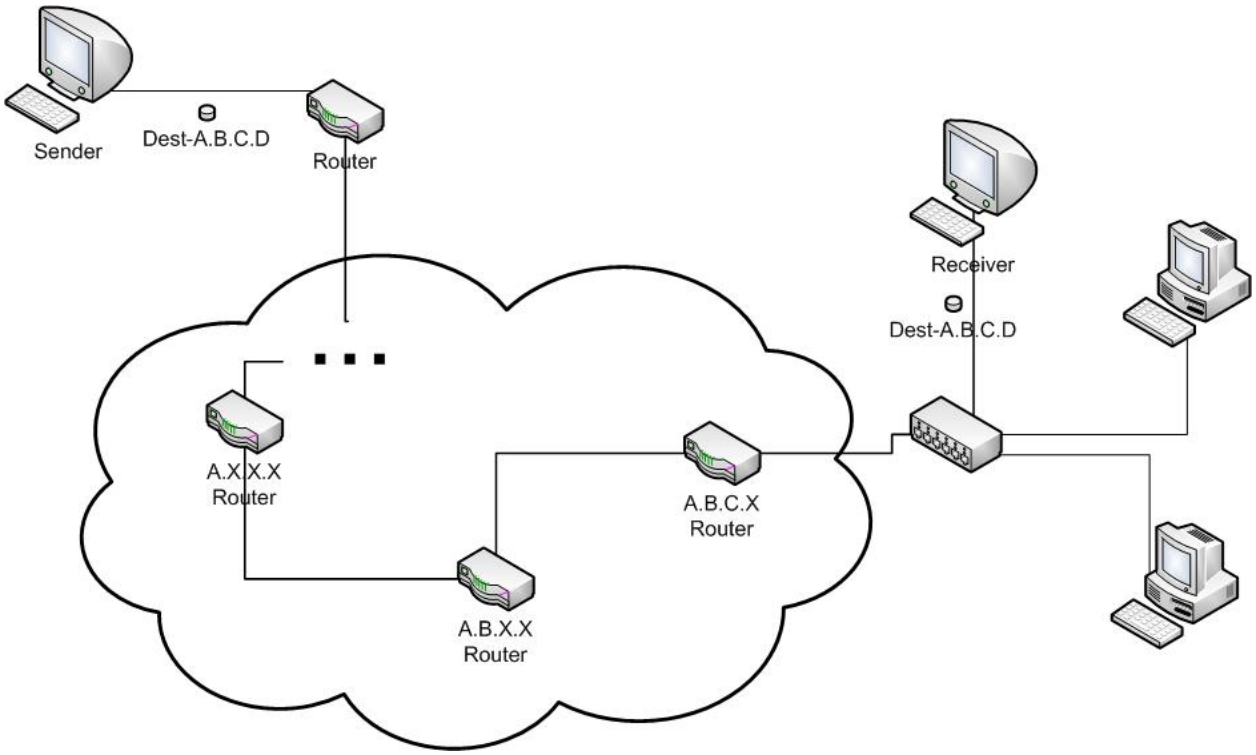
2. MOBILE INTERNET ACCESS

Η σωστή δρομολόγηση δεδομένων μέσω του Internet μπορεί να επιτευχθεί αν και μόνον αν ο κάθε κόμβος μπορεί να αναγνωριστεί μοναδικά σε όλο το Διαδικτυακό χώρο. Αυτό προϋποθέτει από κάθε κόμβο να διαθέτει ένα χαρακτηριστικό που θα τον κάνει μοναδικό. Σε δίκτυα που χρησιμοποιούν το Internet Protocol μια διεύθυνση IP είναι ένας μοναδικός αριθμός που χρησιμοποιείται από συσκευές για τη μεταξύ τους αναγνώριση και συνεννόηση. Κάθε συσκευή που ανήκει στο δίκτυο πρέπει να έχει τη δική της μοναδική διεύθυνση. Μία διεύθυνση IP μπορεί να θεωρηθεί το αντίστοιχο μιας διεύθυνσης κατοικίας ή ενός αριθμού τηλεφώνου. Όπως κάθε διεύθυνση κατοικίας και αριθμός τηλεφώνου αντιστοιχούν σε ένα και μοναδικό κτίριο ή τηλέφωνο, μια IP address χρησιμοποιείται για τη μοναδική αναγνώριση ενός υπολογιστή ή άλλης συσκευής που συνδέεται στο δίκτυο (Εικόνα 2). Το Internet Protocol [4] που επικυρώθηκε από την IETF ορίζει πως η διεύθυνση όχι μόνο προσδιορίζει μοναδικά κάθε κόμβο, αλλά και καθορίζει το σημείο του σύνδεσης του με το δίκτυο. Συνεπώς ο κόμβος πρέπει να βρίσκεται στο δίκτυο που υποδεικνύεται από την IP διεύθυνση του προκειμένου να λαμβάνει τα πακέτα που προορίζονται για αυτόν. Αλλιώς οποιαδήποτε πακέτα έχουν σαν προορισμό τον κόμβο αυτό θα χάνονται.

Έτσι λοιπόν προκειμένου ένας κινητός κόμβος να αλλάξει σημείο σύνδεσης με το δίκτυο, χωρίς όμως να χάσει την ικανότητα του να επικοινωνεί πρέπει να του ανατεθεί μια νέα διεύθυνση. Όπως περιγράφεται λοιπόν στο [5], κάθε κινητός κόμβος (MN) αναγνωρίζεται πάντα από την Home Address (HoA), ανεξαρτήτως από την θέση του στο δίκτυο. Όταν ο κόμβος βρίσκεται μακριά από το σπίτι του, του ανατίθεται επιπλέον μια Care-of-Address (CoA) η οποία προσδιορίζει το ισχύον σημείο σύνδεσης του με το Internet. Το IP πρωτόκολλο παρέχει μηχανισμούς καταχώρησης της Care-of-Address με το Home Agent (HA). Από αυτό το σημείο και μετά οποιαδήποτε πακέτα προορίζονται προς τον MN, προωθούνται από τον HA μέσω τούνελ στην CoA και τελικά παραλαμβάνονται από τον MN.

Η διαχείριση της κινητικότητας των κόμβων σε ένα IP δίκτυο αποτελείται από δύο λειτουργίες. Η πρώτη αφορά την διαχείριση της IP διεύθυνσης, δηλαδή το κατά πόσο

η διεύθυνση είναι τοπολογικά σωστή και κατά πόσο ο Home Agent είναι ενήμερος για οποιαδήποτε αλλαγή σε αυτήν. Η δεύτερη λειτουργία ασχολείται με αν πληρούνται διάφορες προϋποθέσεις ώστε ο κινητός κόμβος να αφήσει την παλιά του διεύθυνση και να κάνει handover. Περισσότερες λεπτομέρειες για τα handover δίδονται στην επόμενη παράγραφο.



Εικόνα 2: Δρομολόγηση στο Internet

2.1. Handovers

Ας δούμε όμως πιο αναλυτικά τι σημαίνει ένα handover. Στις κινητές επικοινωνίες, ο όρος handover αναφέρεται στη διαδικασία μεταφοράς μιας τρέχουσας κλήσης ή δεδομένων από ένα κανάλι του κεντρικού δικτύου σε ένα άλλο. Πολλές φορές στη βιβλιογραφία χρησιμοποιείται και ο όρος handoff.

Στις τηλεπικοινωνίες μπορούν να υπάρξουν διάφοροι λόγοι για τους οποίους μπορεί να γίνει ένα handover:

- όταν ο κόμβος απομακρύνεται από την περιοχή κάλυψης του στοιχείου (πχ κεραία κινητής τηλεφωνίας, 802.11 access point κτλ) που είναι συνδεδεμένος και μπαίνει στην περιοχή κάλυψης κάποιου άλλου η σύνδεση μεταφέρεται στο δεύτερο στοιχείο προκειμένου να αποφευχθεί ο τερματισμός της σύνδεσης λόγω κακού σήματος.
- όταν υπάρχουν επικαλυπτόμενες περιοχές κάλυψης 2 ή περισσοτέρων AP και σε κάποιο από αυτά έχει επιτευχθεί ο μέγιστος αριθμός συνδέσεων, τότε μια νέα προσπάθεια σύνδεσης θα προκαλέσει handover σε κάποιο γειτονικό AP ώστε να εξομαλυνθεί η κίνηση.
- Σε μη-CDMA (Code Division Multiple Access) δίκτυα όταν σε κάποιο κανάλι που ήδη χρησιμοποιείται δημιουργούνται παρεμβολές λόγο κάποιας άλλης σύνδεσης στην ίδια ή σε κάποια γειτονική κυψέλη, η κλήση μεταφέρεται σε ένα διαφορετικό κανάλι στο ίδιο κύτταρο ή σε ένα διαφορετικό κανάλι σε ένα άλλο κύτταρο προκειμένου να αποφευχθούν οι παρεμβολές.
- πάλι σε μη-CDMA δίκτυα όταν η συμπεριφορά κάποιου χρήστη αλλάζει, π.χ. όταν ένας γρήγορα κινούμενος χρήστης, που συνδέεται με ένα μεγάλο, τύπου ομπρέλας κυττάρου, σταματήσει να κινείται, τότε μπορεί να κάνει handover σε ένα μικρότερο κύτταρο προκειμένου αφενός να ελευθερώσει κανάλια στο μεγάλο κύτταρο για άλλους γρήγορα κινούμενους χρήστες και αφετέρου να μειώσει την πιθανή παρεμβολή σε γειτονικά κύτταρα ή χρήστες (αυτή η περίπτωση μπορεί να γίνει και αντίστροφα επίσης: όταν η ταχύτητα κίνησης ενός χρήστη ξεπερνά κάποιο κατώφλι, η σύνδεση του μπορεί να μεταφερθεί στο μεγαλύτερου τύπου κυττάρου προκειμένου να ελαχιστοποιηθεί η συχνότητα των handovers λόγω αυτής της μετακίνησης)
- σε CDMA δίκτυα ένα soft handover (βλ. περισσότερα παρακάτω) μπορεί να προκληθεί προκειμένου να μειώσει την παρέμβαση σε ένα μικρότερο γειτονικό κύτταρο λόγω της near-far[1] επίδρασης ακόμα και αν η σύνδεση του χρήστη έχει πολύ καλά ποιοτικά χαρακτηριστικά.

[1] Το πρόβλημα είναι το εξής: θεωρήστε δέκτη και δύο πομπούς (ο ένας κοντά στο δέκτη, ο άλλος μακριά). Εάν και οι πομποί μεταδώσουν ταυτόχρονα και με ίδια ισχύ, τότε λόγω του νόμου του αντίστροφου τετραγώνου ο δέκτης θα λάβει περισσότερη ισχύ από τον

- Κτλ

Παρακάτω θα δούμε πως τα handovers κατηγοριοποιούνται.

2.1.1. Inter-cell και intra-cell handovers

Στην πιο βασική μορφή handover η σύνδεση ενός κόμβου σε ένα κανάλι ανακατευθύνεται από την τρέχουσα κυψέλη (καλούμενη *αφετηρία*) σε μία άλλη (καλούμενη *προορισμός*) σε ένα νέο κανάλι. Η αφετηρία και ο προορισμός μπορούν είτε να είναι διαφορετικές, είτε να ταυτίζονται. Ένα handover στο οποίο η αφετηρία και ο προορισμός είναι διαφορετικές κυψέλες (ακόμα κι αν είναι στην ίδια περιοχή κυττάρων) καλείται *inter-cell handover*. Ο σκοπός του inter-cell handover είναι να διατηρήσει την σύνδεση δεδομένου ότι ο κόμβος κινείται κατά μήκος περιοχών κάλυψης διαφόρων κυψέλων. Στην περίπτωση όπου η αφετηρία και ο προορισμός συμπίπτουν τότε αλλάζει μόνο το χρησιμοποιημένο κανάλι. Ένα τέτοιο handover καλείται *intra-cell handover*. Ο σκοπός του intra-cell handover είναι να αλλάξει ένα κανάλι, το οποίο μπορεί να προκαλεί παρεμβολές σε ένα άλλο κανάλι.

2.1.2. Soft και Hard Handovers

Εκτός από την ανωτέρω ταξινόμηση τα handover μπορούν επίσης να διαιρεθούν σε soft και hard handovers:

- Hard handover έχουμε όταν ο κόμβος πρώτα αποσυνδέεται από το κανάλι της αφετηρίας έπειτα συνδέεται με το κανάλι του προορισμού. Κατά συνέπεια η σύνδεση διακόπτεται πριν γίνει η σύνδεση στον προορισμό. Για αυτό τον λόγο τέτοια handovers λέγονται επίσης και *break-before-make*. Τα hard handovers

---

κοντινότερο πομπό. Αυτό καθιστά τον μακρινό πομπό αδύνατο "να καταλάβει". Δεδομένου ότι το σήμα του ενός είναι θόρυβος για τον άλλο, το SNR του μακρινού πομπού είναι μικρό.

προορίζονται να είναι στιγμιαία προκειμένου να ελαχιστοποιηθεί η διάσπαση της κλήσης.

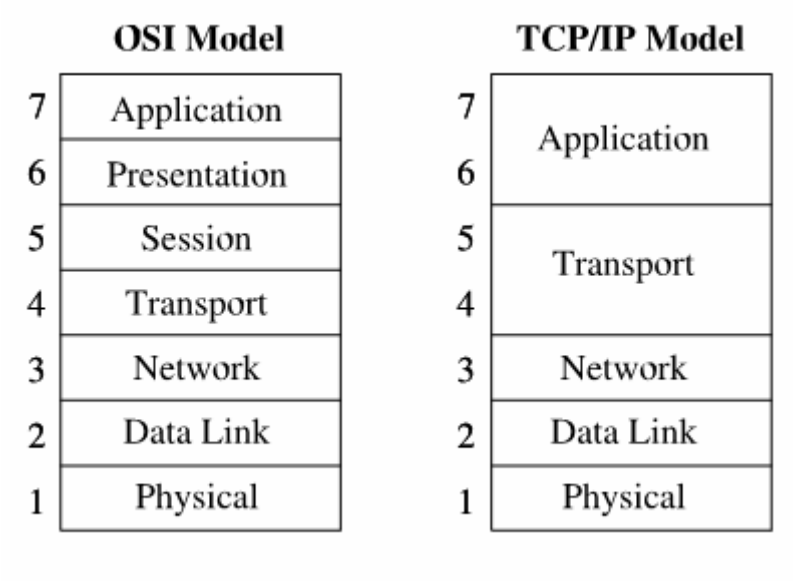
- Soft handovers πραγματοποιούνται όταν το κανάλι της αφετηρίας διατηρείται προσωρινά και χρησιμοποιείται παράλληλα με το κανάλι του προορισμού. Σε αυτήν την περίπτωση η σύνδεση στον προορισμό πραγματοποιείται προτού διακοπεί η σύνδεση στην αφετηρία, ως εκ τούτου αυτό το handover αποκαλείτια και *make-before-break*. Το διάστημα, κατά τη διάρκεια του οποίου οι δύο συνδέσεις χρησιμοποιούνται παράλληλα, μπορεί να είναι μικρό ή στιγμιαίο. Για αυτόν τον λόγο το soft handover θεωρείται σαν κατάσταση της κλήσης, παρά σαν ένα γεγονός. Ένα soft handover μπορεί να περιλαμβάνει τη χρησιμοποίηση περισσοτέρων από δύο συνδέσεων, π.χ. οι συνδέσεις σε τρία, τέσσερα ή περισσότερα κύτταρα μπορούν να διατηρηθούν συγχρόνως. Όταν μια σύνδεση είναι σε κατάσταση soft handover το σήμα του καλύτερου όλων των χρησιμοποιημένων καναλιών μπορεί να χρησιμοποιηθεί για την κλήση σε μια δεδομένη στιγμή (Selection Diversity) ή όλα τα σήματα μπορούν να συνδυαστούν για να παραγάγουν ένα σαφέστερο αντίγραφο του σήματος (Maximal Ratio Combining Diversity). Το τελευταίο είναι πιο επωφελής, και ειδικά όταν εκτελείται τέτοιος συνδυασμός και στο downlink (forward link) και στο uplink (reverse link) το handover καλείται *softer*.

Ένα πλεονέκτημα των Hard handovers είναι ότι σε κάθε χρονική στιγμή μια σύνδεση χρησιμοποιεί μόνο ένα κανάλι. Τα hard handovers είναι συνήθως πολύ μικρά σε διάρκεια και σπανίως αντιληπτά από το χρήστη σε περίπτωση φωνής ή multimedia δεδομένων. Ένα άλλο πλεονέκτημα των hard handovers είναι ότι δεν απαιτεί επιπλέον hardware και άρα καθιστά τις συσκευές πιο φτηνές και πιο απλές. Ένα μειονέκτημα είναι όμως ότι εάν ένα handover αποτύχει μπορούν να προκληθούν σοβαρές παρεμβολές ή ακόμα και να τερματιστεί η σύνδεση. Οι τεχνολογίες, που χρησιμοποιούν hard handovers, είναι εφοδιασμένες συνήθως με κατάλληλες διαδικασίες που μπορούν να επανεγκαθιδρύσουν τη σύνδεση στο κύτταρο αφετηρίας εάν η σύνδεση στο κύτταρο προορισμού δεν μπορεί να γίνει. Παρόλα αυτά η επανεγκαθίδρυση αυτής της σύνδεσης μπορεί μην είναι πάντα δυνατή οπότε σε αυτή την περίπτωση η κλήση θα τερματιστεί.

Ένα πλεονέκτημα των Soft handovers είναι ότι η σύνδεση στο κύτταρο αφετηρίας τερματίζεται μόνο όταν καθιερωθεί μια αξιόπιστη σύνδεση στο κύτταρο προορισμού και επομένως οι πιθανότητα ότι η κλήση να τερματιστεί λόγω αποτυχημένου handover είναι μικρότερη. Εντούτοις, ένα πολύ σοβαρό πλεονέκτημα προέρχεται από το γεγονός ότι διατηρούνται ταυτόχρονα συνδέσεις σε πολλαπλά κύτταρα και συνεπώς η σύνδεση θα μπορούσε να αποτύχει μόνο εάν όλα τα κανάλια εξασθενίζουν συγχρόνως. Η εξασθένιση και οι παρεμβολές σε διαφορετικά κανάλια είμαι στατιστικά ανεξάρτητες και επομένως η πιθανότητα για όλα τα κανάλια μαζί είναι πολύ χαμηλή. Κατά συνέπεια η αξιοπιστία της σύνδεσης γίνεται υψηλότερη όταν χρησιμοποιούνται soft handovers. Επειδή σε ένα κυψελοειδές δίκτυο η πλειοψηφία των handovers εμφανίζεται σε περιοχές χαμηλής κάλυψης, όπου οι κλήσεις γίνονταν συχνά αναξιόπιστες καθώς το κανάλι εξασθενίζει, τα soft handovers φέρνουν μια σημαντική βελτίωση στην αξιοπιστία των κλήσεων κάνοντας την εξασθένιση ενός καναλιού μη κρίσιμο παράγοντα. Αυτό το πλεονέκτημα έρχεται με κόστος του πιο σύνθετου hardware στη συσκευή, η οποία πρέπει να είναι ικανή να επεξεργάζεται διάφορα κανάλια παράλληλα. Ένα άλλο κόστος των soft handovers είναι χρήση πολλαπλών καναλιών στο δίκτυο για να υποστηριχτεί μόνο μια κλήση. Αυτό μειώνει τον αριθμό των ελεύθερων καναλιών και άρα και την χωρητικότητα του δικτύου. Με τη ρύθμιση της διάρκειας των soft handovers και του μεγέθους των περιοχών, στις οποίες εμφανίζονται, το όφελος της πρόσθετης αξιοπιστίας κλήσης μπορεί να εξισορροπήσει το κόστος της μειωμένης ικανότητας.

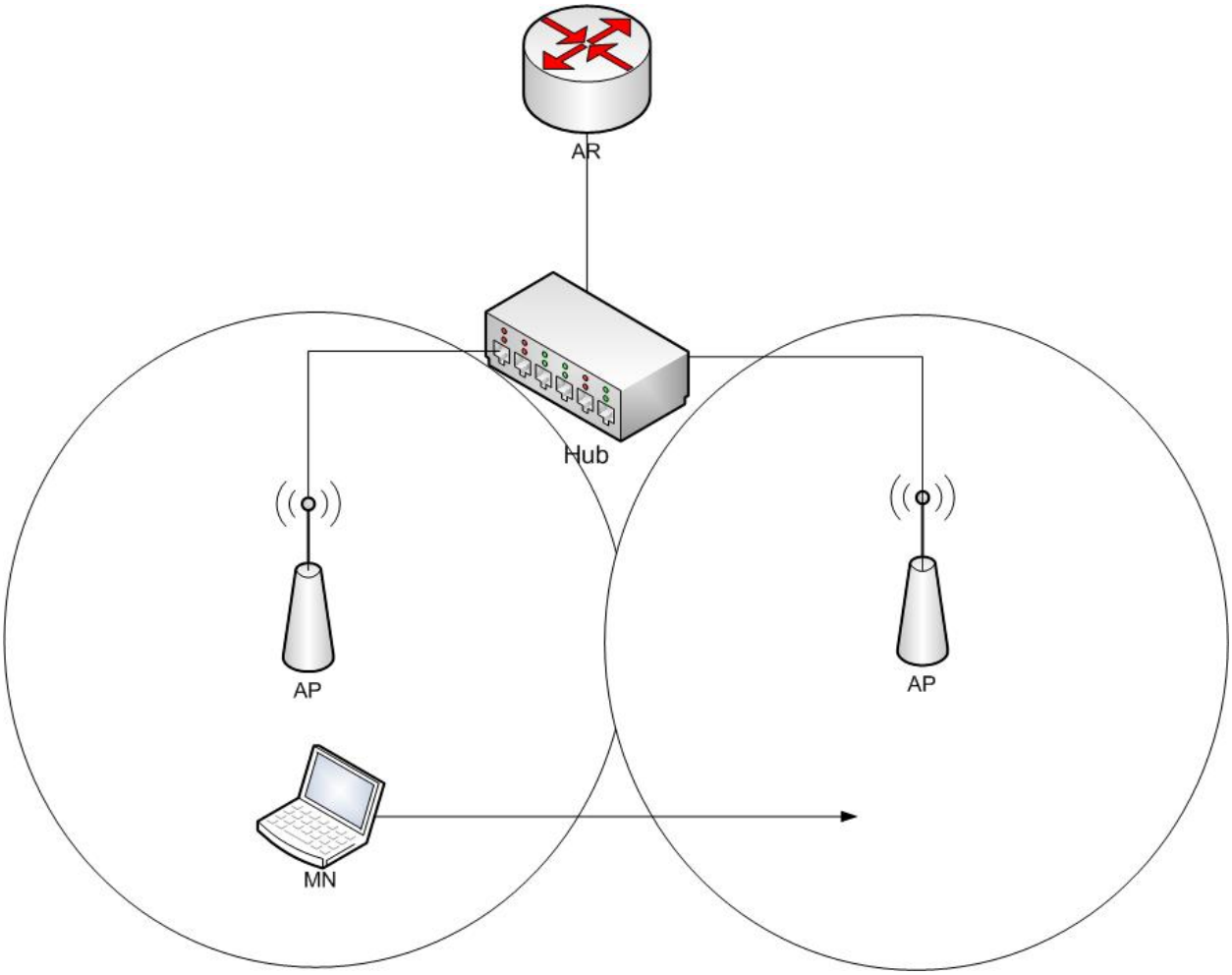
2.1.3. Layer-2 και Layer-3 Handovers

Το 1977 ο Διεθνής Οργανισμός Τυποποιήσεων ISO (International Organization for Standardization) ανακοίνωσε το πρότυπο ¨Πρότυπο Διασύνδεσης Ανοικτών Συστημάτων¨ OSI (Open System Interconnection reference model). Το OSI αποτελεί πλαίσιο μέσα στο οποίο κινούνται οι λεπτομερείς πλέον τυποποιήσεις, για την επίλυση όλων των προβλημάτων που εμφανίζονται στις επικοινωνίες υπολογιστών διαφορετικών κατασκευαστών.



Εικόνα 3: OSI model και TCP/IP stack

Πάνω στο μοντέλο OSI έχει πια τυποποιηθεί και το πρωτόκολλο TCP/IP του οποίου η στοίβα παρουσιάζεται στην Εικόνα 3. Στο πρωτόκολλο TCP/IP το επίπεδο Διασύνδεσης Δεδομένων (L2) ορίζει το hardware του δικτύου, διαχειρίζεται τις συνδέσεις και προωθεί δεδομένα από το φυσικό επίπεδο στο επίπεδο 3. Το επίπεδο Δικτύου (L3) ασχολείται με διαδικασίες διευθυνσιοδότησης και δρομολόγησης πακέτων. Πιο απλά καθορίζει που να σταλούν τα πακέτα σύμφωνα με τις πληροφορίες που έχει. Τα handovers λοιπόν σε IP δίκτυα γίνονται είτε στο δεύτερο επίπεδο (L2 handover), είτε στο τρίτο επίπεδο (L3 handover). Ένα Data Link Layer (L2) handover συμβαίνει όταν ένας κινητός κόμβος κινείται μεταξύ δύο ή περισσοτέρων Access Points (AP) που είναι συνδεδεμένα στην ίδια διεπαφή ενός Access Router (AR) μέσω ενός switch ή ενός hub, όπως φαίνεται στην Εικόνα 4. Ένα τέτοιο handover περιλαμβάνει μόνο διαδικασίες επιπέδου διασύνδεσης, όπως πχ η επαναπιστοποίηση με ένα σταθμό βάσης IEEE 802.11b. Αντίθετα σε ένα L3 handover το σημείου σύνδεσης στο δίκτυο αλλάζει σε ένα διαφορετικό υποδίκτυο, σαν αποτέλεσμα αλλαγής AR (Εικόνα 5), ή αλλαγή διεπαφής στον ίδιο AR. Συνεπώς ανακτάται μια καινούρια IP διεύθυνση η οποία γίνεται η CoA διεύθυνση του κόμβου.



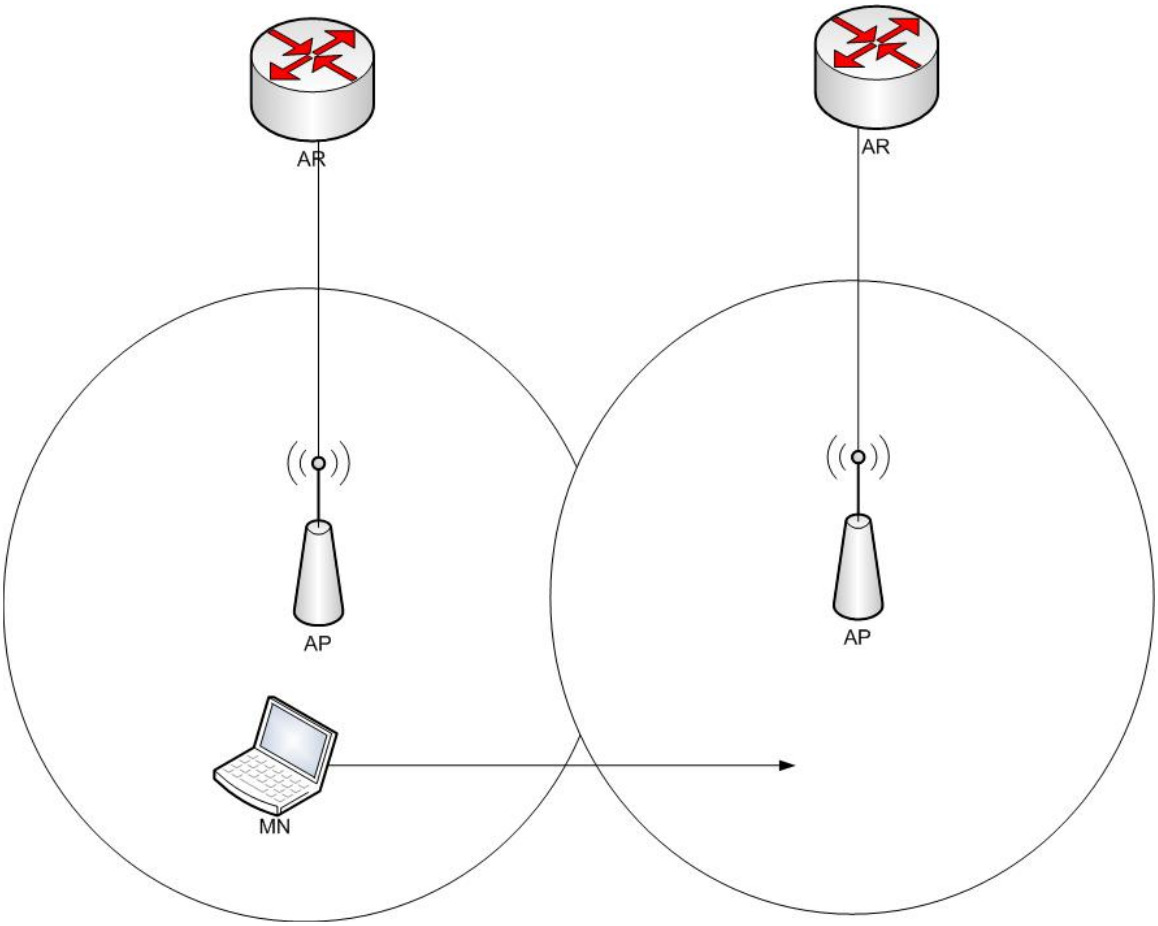
Εικόνα 4: Data Link Layer handover

Ένα L3 handover πάντα συνεπάγεται ένα L2 handover, ενώ το αντίθετο δεν ισχύει πάντα. Μια λεπτομερέστατη αναφορά στα handover που θα ασχοληθούμε παρουσιάζεται στο κεφάλαιο 3.

2.2. Mobility Support στο IPv4

Σύμφωνα με το [5] η διεύθυνση IP όπως είδαμε και σε προηγούμενο εδάφιο αποτελείται από δύο μέρη: Το πρώτο καθορίζει το δίκτυο στο οποίο ανήκει ο κόμβος, ενώ το δεύτερο καθορίζει τον αριθμό του κόμβου στο υποδίκτυο. Έτσι λοιπόν το πρωτόκολλο IP αποφασίζει το επόμενο hop σύμφωνα με την IP διεύθυνση του προορισμού. Παράλληλα τα υψηλότερα επίπεδα, όπως το TCP, διατηρούν

πληροφορίες για τις ενεργές συνδέσεις, αποτελούμενες από την τετράδα iP/port αφετηρίας, προορισμού. Κατά συνέπεια, κατά την προσπάθεια υποστήριξης κινούμενων κόμβων στο Διαδίκτυο κάτω από τα υπάρχουσα πρωτοκόλλα, βρέθηκαν αντιμέτωποι με δύο αμοιβαία συγκρουόμενες απαιτήσεις: (1) ένας κινητός κόμβος πρέπει να αλλάξει τη διεύθυνση IP του όποτε αλλάζει το σημείο σύνδεσής του, έτσι ώστε τα πακέτα που προορίζονται στον κόμβο να καθοδηγούνται σωστά, (2) για να διατηρηθούν οι υπάρχουσες TCP συνδέσεις, ο κινητός κόμβος πρέπει να κρατήσει ίδια τη διεύθυνση IP του, καθώς αλλαγή της IP διεύθυνσης θα τερματίσει την σύνδεση.



Εικόνα 5: Network Layer handover

Το Mobile IPv4 [3], όπως προτάθηκε από την IETF, είναι σχεδιασμένο να λύνει αυτό το πρόβλημα επιτρέποντας σε κάθε κινητό κόμβο να έχει δύο IP διευθύνσεις και διατηρώντας διαφανώς μία σύνδεση μεταξύ τους. Η μία διεύθυνση είναι η μόνιμη Home Address (HoA) που ορίζεται στο home network και χρησιμοποιείται στον

καθορισμό endpoints επικοινωνίας. Η άλλη διεύθυνση, η λεγόμενη προσωρινή Care-of-Address (CoA), αντιπροσωπεύει την τρέχουσα θέση του κόμβου. Οι κύριοι στόχοι του Mobile IP είναι να κατασταθεί η κινητικότητα διαφανής στα πρωτόκολλα υψηλότερων επιπέδων, με ταυτόχρονη ελαχιστοποίηση των αλλαγών στην υπάρχουσα υποδομή.

Αυτή η σύνδεση μεταξύ HoA και CoA που αναφέραμε παραπάνω διατηρείται από μερικούς εξειδικευμένους δρομολογητές γνωστούς ως mobility agents. Οι mobility agents είναι δύο τύπων – Home Agents (HA) και Foreign Agents (FA).

Ο home agent, ένας καθορισμένος δρομολογητής στο home network του κινητού κόμβου, διατηρεί την σύνδεση των διευθύνσεων σε ένα πίνακα, τον λεγόμενο mobility binging table, όπου κάθε εγγραφή ορίζεται από την τριάδα, < Home Address, Care of Address, Lifetime >. Η εικόνα 6 παρουσιάζει έναν τέτοιο πίνακα. Σκοπός αυτού του πίνακα είναι να δέσει την Home Address ενός κινητού κόμβου με την Care of Address ώστε να διαβιβαστούν τα πακέτα αναλόγως.

Home Address	Care-of Address	Lifetime (in sec)
131.193.171.4	128.172.23.78	200
131.193.171.2	119.123.56.78	150

Εικόνα 6: Mobility binding table

Οι foreign agents είναι ειδικευμένοι δρομολογητές στο foreign network όπου ο κινητός κόμβος βρίσκεται αυτήν την περίοδο. Ο foreign agent διατηρεί μία λίστα, την λεγόμενη visitor list, που περιέχει πληροφορίες για κινητούς κόμβους που επισκέπτονται αυτή την περίοδο το ξένο δίκτυο. Κάθε εγγραφή στη λίστα προσδιορίζεται από τετράδα < Home Address, Home Agent Address, MAC Address, Lifetime >, όπως φαίνεται στην εικόνα 7.

Home Address	Home Agent Address	Media Address	Lifetime (in s)
131.193.44.14	131.193.44.7	00-60-08-95-66-E1	150
131.193.33.19	131.193.33.1	00-60-08-68-A2-56	200

Εικόνα 7: Visitor List

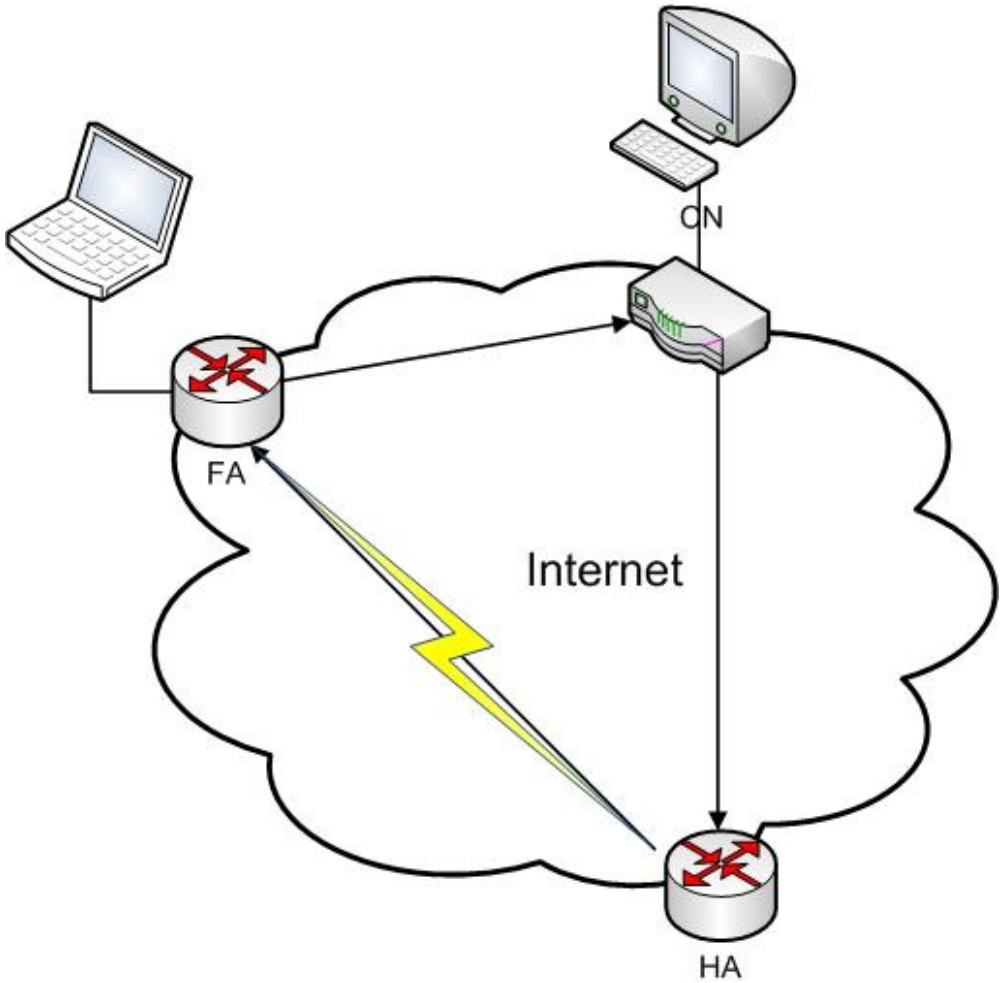
Σε ένα χαρακτηριστικό σενάριο, η CoA ενός κινητού κόμβου είναι η IP διεύθυνση του foreign agent. Μπορεί να υπάρξει και ένα άλλο είδος CoA, γνωστή ως collocated CoA, (cocoa), η οποία λαμβάνεται συνήθως από κάποιο εξωτερικό μηχανισμό διευθυνσιοδότησης, όπως ο Dynamic Host Configuration Protocol (DHCP).
Σύμφωνα λοιπόν με όλα αυτά η επικοινωνία με έναν κόμβο που βρίσκεται εκτός από το home network του γίνεται ως εξής:

- Όταν ο correspondent node (CN) θέλει να επικοινωνήσει με τον κινητό κόμβο (MN), στέλνει ένα πακέτο στη μόνιμη IP διεύθυνση (HoA) του κινητού κόμβου.
- Ο home agent αναχαιτίζει το πακέτο και ελέγχει το mobility binding table για να δει εάν ο κινητός κόμβος επισκέπτεται αυτήν την περίοδο κάποιο άλλο δίκτυο.
- Ο home agent βρίσκει την CoA του κινητού κόμβου και κατασκευάζει ένα νέο IP πακέτο που περιέχει την CoA του MN σαν διεύθυνση προορισμού του πακέτου. Το παλιό πακέτο εμφωλεύεται στο νέο και έπειτα δρομολογείται. Αυτή η διαδικασία ονομάζεται IP within IP encapsulation [8].
- Όταν το πακέτο φθάσει στο τρέχον δίκτυο του κινητού κόμβου, ο foreign agent εξάγει το αρχικό πακέτο και ανακαλύπτει την HoA του κινητού κόμβου. Συμβουλεύεται έπειτα την Visitor List για να δει εάν έχει κάποια εγγραφή για εκείνο τον κινητό κόμβο.
- Εάν υπάρχει εγγραφή ο foreign agent ανακτά την MAC διεύθυνση του κόμβου και του προωθεί το πακέτο.

- Όταν ο κινητός κόμβος θέλει να στείλει ένα μήνυμα σε έναν correspondent κόμβο, διαβιβάζει το πακέτο στον FA, το οποίο αναμεταδίδει στη συνέχεια στον CN χρησιμοποιώντας την κανονική IP δρομολόγηση.

Η τεχνική του Mobile IP έλυσε μεν ένα πολύ σημαντικό πρόβλημα, άφησε όμως πίσω της ένα πολύ σημαντικό θέμα, την τριγωνική δρομολόγηση (triangular routing).

Η βασική ιδέα πίσω από την triangular routing είναι η ακόλουθη: Ένας κόμβος στέλνει ένα πακέτο σε έναν κινητό κόμβο που είναι στο ίδιο δίκτυο. Τυχαίνει όμως ο home agent του κινητού κόμβου να είναι πολύ μακριά, στην άλλη ¨μεριά¨ του Διαδικτύου. Έτσι ο CN απευθύνει όλα τα πακέτα στο home network, περνούν δηλαδή διαμέσου όλου του Διαδικτύου για να φθάσουν στον home agent και έπειτα δρομολογούνται πάλι πίσω μέσω τούνελ στον foreign agent, ο οποίος τελικά τα προωθεί στον MN (Εικόνα 8).



Εικόνα 8: Triangular Routing

Αυτή η προσέγγιση έχει αρκετά μειονεκτήματα. Η real-time κίνηση από εφαρμογές όπως video conference και Voice over IP (VoIP) απαιτούν σφιχτά όρια όσο αναφορά την end to end καθυστέρησης και απώλειας πακέτων. Η τριγωνική δρομολόγηση θα αυξήσει την end to end καθυστέρηση από τον CN στον MN καθώς η δρομολόγηση δεν είναι βέλτιστη. Επίσης ένα L3 handover περιλαμβάνει την απόκτηση μιας CoA και την ενημέρωση των mobility bindings των CNs και HA, και συνεπώς εισάγει μια επιπλέον καθυστέρηση εκτός από την καθυστέρηση του L2 handover. Αποτέλεσμα της handover καθυστέρησης είναι η διακοπή των εγκαθιδρυμένων συνδέσεων προς στιγμήν, άρα και η απώλεια πακέτων και τελικά την απώλεια ποιότητας του multimedia stream. Η Mobile IP τεχνική είναι επίσης πολύ ανεπαρκής όταν αναλογιστούμε το overhead από το tunneling κάθε πακέτου που λαμβάνεται όταν ο MN είναι εκτός του home network του.
Θα ήταν βέλτιστο εάν ο CN μπορούσε να ανακαλύψει ότι ο κινητός κόμβος είναι στο ίδιο δίκτυο και παραδίδει το πακέτο άμεσα. Στόχος είναι να παραδοθούν τα πακέτα όσο το δυνατόν γρηγορότερα. Δηλαδή αρκεί τα πακέτα του CN προς τον MN να δρομολογηθούν κατευθείαν στην CoA του MN, χωρίς να χρειαστεί να περάσουν από τον HA. Ο Perkins και Johnson πρότειναν στο [10] την τεχνική του Route Optimization, η οποία θα έλυνε αυτό το πρόβλημα, αλλά πότε δεν καθιερώθηκε από την IETF, καθώς οι προσπάθειες είχαν ήδη επικεντρωθεί στην νέα έκδοση του πρωτοκόλλου IP, την IPv6.

2.3. Mobility Support στο IPv6

Για σχεδόν 30 χρόνια το πρωτόκολλο IP, αποδείχτηκε ικανό να αντιμετωπίσει την αλματώδη ανάπτυξη του Διαδικτύου. Προβλήματα που ήταν αδύνατο να προβλεφτούν την δεκαετία του ’80, αντιμετωπίστηκαν ικανοποιητικά, επεκτείνοντας το αρχικό πρωτόκολλο. Δίκαια θεωρείται ως το πιο πετυχημένο πρωτόκολλο, καθώς παρά την ηλικία του κατάφερε να διασυνδέσει εκατομμύρια συστήματα διαφορετικών αρχιτεκτονικών. Η μεγάλη όμως ανάπτυξη του Διαδικτύου, καθώς και οι απαιτήσεις των νέων δικτυακών εφαρμογών δεν μπορούν να αντιμετωπισθούν από το IPv4.

Έτσι η IETF μετά από πολλές προτάσεις για το IPng (Next Generation Internet Protocol), κατέληξε στην δημιουργία ενός νέου πρωτοκόλλου στα χνάρια του IP και έτσι το 1998 παρουσιάστηκε η 6η έκδοση του πρωτοκόλλου IP, με την ονομασία IPv6 [11].

2.3.1. Από το IPv4 στο IPv6

Το IPv6 δημιουργήθηκε όπως είπαμε για να λύσει τους έμφυτους περιορισμούς του πρωτοκόλλου IPv4. Ο πιο διαδεδομένος περιορισμός ήταν ο άδικος διαμοιρασμός των καθολικών IP διευθύνσεων, που ευνοούσαν ιδιαίτερα την Αμερική. Παραδείγματος χάριν το πανεπιστημίου του Stanford στην Αμερική διαθέτει περισσότερες δημόσιες IP διευθύνσεις από ολόκληρη την Κίνα. Η έλλειψη διευθύνσεων θα δημιουργούσε ένα πού σημαντικό πρόβλημα, που όμως αντιμετωπίστηκε το NAT (Network Address Translation), το οποίο υπόσχεται να επεκτείνει την IPv4 από τα 32bit στα 48. Παρόλο που το NAT επιτρέπει σε περισσότερους ανθρώπους να συνδεθούν στο internet, όπως επίσης επιτρέπει σε μικρούς οργανισμούς να διαμορφώσουν μόνοι τους το δικό τους χώρο διευθύνσεων, χωρίς να βασίζονται στις αρμόδιες αρχές να τους δώσουν μοναδικές διευθύνσεις, αποτυγχάνει να παρέχει το καθολική δρομολόγηση. Έτσι αποκλείει κόμβους από λειτουργία ως server, ή την χρήση peer-to-peer εφαρμογών. Για να υπερνικηθεί αυτό το πρόβλημα ένας κεντρικός υπολογιστής απαιτείται για να διαιτητεύσει μεταξύ των client, και συνεπώς το δίκτυο σταματά να είναι peer-to-peer. Επίσης το NAT δεν λύνει το πρόβλημα της άδικης κατανομής διευθύνσεων.

Η ανάπτυξη του διαδικτύου στο απώτερο μέλλον όμως δεν θα είναι δυνατή, παρόλη την πνοή που έφερε το NAT. Το IPv6 από την άλλη έχει IP διευθύνσεις των 128 bit, και είναι ήδη διαθέσιμο[2]. Πολλές οργανώσεις έχουν προσπαθήσει να δώσουν επιπλέον ώθηση στο IPv6. Η Ευρωπαϊκή Ένωση έχει επενδύσει πάνω από 20 εκατομμύρια Ευρώ σε ένα IPv6 δίκτυο όπως το 6Net[3] και το 6Diss[4] σε μία

[2] Υπάρχει ήδη πλήρης IPv6 κάλυψη σε πολλές πλατφόρμες του εμπορίου, όπως τα Windows XP, Windows Vista, MacOS, Symbian OS, Linux, openBSD, Sony Playstation, καθώς και σε πολλούς router που ήδη κυκλοφορούν
[3] http://www.6net.org
[4] http://www.6diss.org

προσπάθεια να επιταχυνθεί η μετάβαση στο IPv6. Η 3rd Generation Partnership Project (3GPP) έχει προτείνει η UMTS (Universal Mobile Telecommunications System) Release 5 για IMS (Internet Multimedia Service) να λειτουργεί μόνο σε IPv6 [12, 11]. Χωρίς αμφιβολία η καθολική μετάβαση σε IPv6 είναι αναπόφευκτη.

2.3.2. IPv6

Αναφέρουμε μερικά από τα πλεονεκτήματα του IPv6 σε σχέση με το IPv4. Πολλά από αυτά θα λέγαμε πως δεν είναι απλά πλεονεκτήματα, αλλά άμεσες αναγκαίες αλλαγές στο IP πρωτόκολλο.

- Εκτεταμένη δυνατότητα διευθυνσιοδότησης: Το IPv6 αυξάνει το μέγεθος της επικεφαλίδας από 32 σε 128 bits, προσφέροντας δυνατότητες για περισσότερα επίπεδα διευθυνσιοδότησης, ¨ανεξάντλητο¨ χώρο διευθύνσεων και απλούστερη αυτοδιαμόρφωση των διευθύνσεων (*autoconfiguration*). Η διαβαθμισιμότητα της δρομολόγησης multicast έχει βελτιωθεί, προσθέτοντας το πεδίο *scope* στη διεύθυνση που πληροφορεί το δρομολογητή για την περιοχή των host που ¨ακούνε¨(π.χ. LAN, WAN, internet).
- Καθολική μοναδική ιεραρχική διευθυνσιοδότηση: Η διευθυνσιοδότηση βασίζεται σε prefixes και όχι κλάσεις, προσφέροντας έτσι καλύτερη ταξινόμηση των κόμβων, μικρότερα routing tables και αποδοτικότερη δρομολόγηση στο δίκτυο κορμού[5]
- Υποστήριξη ενθυλάκωσης: Υποστηρίζεται ενθυλάκωση στο IPv6 πακέτο άλλων πρωτοκόλλων καθώς και του ίδιου του IPv6.
- Απλοποιημένη επικεφαλίδα: Ορισμένα πεδία του IPv4 απουσιάζουν από το IPv6 ή έχουν γίνει προαιρετικά. Αυτό βοηθά στη μείωση του κόστους δρομολόγησης για κάθε πακέτο και του κόστους σε εύρος ζώνης που καταναλώνει η επικεφαλίδα. Η επικεφαλίδα, επίσης, έχει σταθερό μήκος, και

[5] Αντί των πάνω από 2 εκατομμυρίων μονοπατιών σε core routers για το IPv4, τώρα έχουμε το μέγιστο 8.192, όπως παίρνουμε από τα 13 bits του TLA πεδίου (Top Level Aggregation) στην IPv6 διεύθυνση

όπως αναφέραμε στο προηγούμενο κεφάλαιο οι δρομολογητές έχουν καλύτερη απόδοση για τέτοιες επικεφαλίδες.

- Βελτιωμένη υποστήριξη για επεκτάσεις και επιλογές της επικεφαλίδας: Το IPv6 διαθέτει υποστήριξη προαιρετικών πεδίων σε ξεχωριστές επικεφαλίδες. Αυτό διευκολύνει την απόδοση της απλής δρομολόγησης, αφού δεν χρειάζεται κάθε δρομολογητής να επεξεργαστεί αυτά τα πεδία, αν κάτι τέτοιο δεν είναι αναγκαίο.
- Έλεγχος ροής στο επίπεδο IP: Μια καινούρια λειτουργία έχει προστεθεί που κατηγοριοποιεί τα πακέτα ενός αποστολέα σε μια συγκεκριμένη ροή(*flow*). Αυτή η ροή μπορεί να αντιμετωπιστεί με κάποιο ειδικό τρόπο(π.χ. μια ροή δεδομένων live streaming video).
- Ασφάλεια στο επίπεδο IP:Το IPv6 προσφέρει, μέσω των επικεφαλίδων επέκτασης, ασφάλεια (Authentication Header) και κρυπτογράφηση δεδομένων (Encapsulated Security Payload).
- Υποστήριξη mobility: Το MIPv6 υλοποιείται βασιζόμενο σε χαρακτηριστικά του IPv6 που είναι ήδη ολοκληρωμένα.
- Παροχή Quality of Service: Παρέχετε η δυνατότητα ταξινόμησης πακέτων σε διάφορες ροές, διαφορετικής προτεραιότητας

2.3.2.1. Address Resolution στο IPv6

Το Address Resolution Protocol (ARP) του TCP/IP είναι ένα γενικό πρωτόκολλο για την δυναμική απεικόνιση Network layer διευθύνσεων, σε Link layer διευθύνσεις. Ακόμα κι αν σχεδιάστηκε για την 4η έκδοση του IP, τα μηνύματα που χρησιμοποιεί επιτρέπουν διευθύνσεις μεταβλητού μήκους και για τα δύο επίπεδα. Αυτή η ευελιξία σημαίνει ότι θα ήταν θεωρητικά δυνατό να χρησιμοποιηθεί αυτό το πρωτόκολλο και για την 6η έκδοση του IP (IPv6). Έτσι με ελάχιστες μετατροπές, θα μπορούσαμε να χρησιμοποιήσουμε το ARP σχεδόν αυτούσιο.

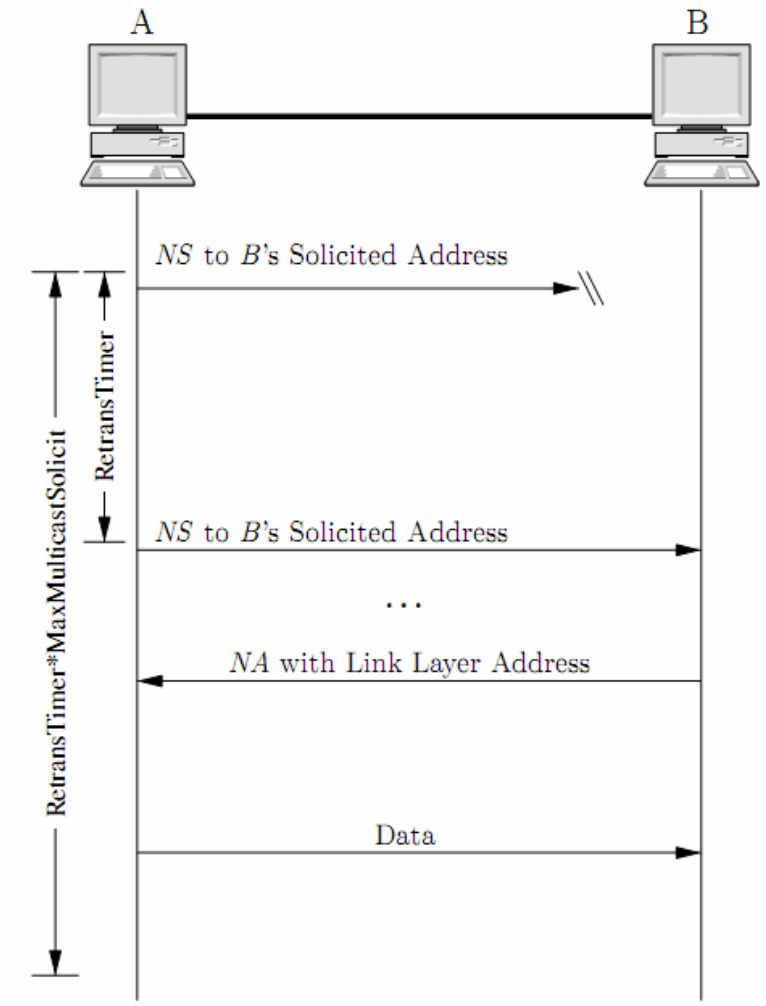
Οι σχεδιαστές του IPv6 επέλεξαν να μην το κάνουν αυτό. Η αλλαγή του IP είναι μια μεγάλη υπόθεση που είναι εν εξελίξει για πολλά έτη, και αντιπροσωπεύει μια σπάνια ευκαιρία να αλλαχτούν οι διάφορες πτυχές του TCP/IP. Έτσι η IETF αποφάσισε να εκμεταλλευθεί τις αλλαγές στο IPv6 και να εξετάσει λεπτομερώς όχι μόνο το ίδιο το IP, αλλά και πολλά από τα πρωτόκολλα που "υποστήριζαν" ή "βοηθούσαν" το IP. Στο IPv6 λοιπόν, το Address resolution έχει συνδυαστεί με διάφορες λειτουργίες που εκτελούνται από το ICMP και με κάποιες επιπλέον λειτουργίες για να δημιουργήσει το Neighbor Discovery Protocol (NDP) [13]. Ο όρος "neighbor" στο IPv6 αναφέρεται απλά σε συσκευές σε ένα τοπικό δίκτυο, και όπως το όνομα υπονοεί, το NDP είναι αρμόδιο για την επικοινωνία μεταξύ των γειτόνων.

Οι βασικές λειτουργίες του NDP δεν διαφέρουν και πάρα πολύ από αυτές του ARP (Εικόνα 9). Η ανάλυση της διεύθυνσης είναι ακόμα δυναμική και βασίζεται στη χρήση cache tables που διατηρούν ζευγάρια IP και MAC διευθύνσεων. Κάθε συσκευή σε ένα φυσικό δίκτυο κρατάει τέτοιες πληροφορίες για τους γείτονές της. Όταν μια συσκευή θέλει να στείλει ένα IPv6 πακέτο δεδομένων σε έναν γείτονα της αλλά δεν έχει τη MAC διεύθυνση του, ξεκινά τη διαδικασία ανάλυσης διεύθυνσης. Στο παρακάτω παράδειγμα ας θεωρήσουμε πως η συσκευή Α προσπαθεί να στείλει στη συσκευή Β.

Αντί της αποστολής ενός *ARP Request* μηνύματος, η Α δημιουργεί ένα *ND Neighbor Solicitation* μήνυμα. Εδώ υπάρχει η πρώτη μεγάλη αλλαγή σε σχέση με το ARP. Εάν το data link πρωτόκολλο που χρησιμοποιείται υποστηρίζει multicasting, όπως πχ το Ethernet, το *Neighbor Solicitation* δεν είναι broadcast μήνυμα. Άντ' αυτού, στέλνεται στη *solicited-node address* της συσκευής της οποίας την IPv6 διεύθυνση προσπαθούμε να αναλύσουμε. Έτσι το Α θα στείλει ένα multicast μήνυμα στη *solicited-node multicast* διεύθυνση [14] της συσκευής Β. Το πακέτο θα περιέχει την link-layer διεύθυνση στο source link-layer address πεδίο.

Η συσκευή Β θα λάβει το *Neighbor Solicitation* και θα απαντήσει με *Neighbor Advertisement*, κάτι ανάλογο του ARP Reply. Το *Neighbor Advertisement* θα περιέχει την link-layer διεύθυνση της συσκευής Β στο target link-layer address πεδίο.

Η συσκευή Α θα περιμένει το *Neighbor Advertisement* της συσκευής Β για μια περίοδο MAX_MULTICAST_SOLICIT[6]*RetransTimer[7] δευτερολέπτων, στέλνοντας νέο *Neighbor Solicitation* κάθε RetransTimer δευτερόλεπτα.



Εικόνα 9: IPv6 Neighbor Discovery

Μέχρι η συσκευή Α να λάβει το *Neighbor Advertisement* τοποθετεί οποιοδήποτε πακέτο προορίζεται για τη συσκευή Β σε ουρά. Μόλις λοιπόν η συσκευή Α λάβει το *Neighbor Advertisement* στέλνει όλα τα πακέτα που βρίσκονται στην ουρά στον Β και προσθέτει την αντιστοίχηση των IP, link-layer διευθύνσεων του Β στην neighbor

[6] Default 3 φορές [13]
[7] Default 1 δευτερόλεπτο [13]

cache της. Από αυτή τη στιγμή και για όποια πακέτα προορίζονται για την συσκευή Β, η Α θα βρίσκει την link-layer διεύθυνση της Β στην neighbor cache της.
Για επιπλέον αποδοτικότητα, υποστηρίζεται cross-resolution [15] όπως και στο IPv4. Αυτό επιτυγχάνεται ενσωματώνοντας την link-layer διεύθυνση της συσκευής Α στο *Neighbor Solicitation.* Έτσι η συσκευή Β θα μπορεί να εγγράψει το ζευγάρι IP, link layer διευθύνσεων της Α στην δικιά της neighbor cache.

2.3.2.2. Autoconfiguration

Στο IPv4 η διευθυνσιοδότηση γινόταν είτε χειροκίνητα, είτε με αυτόματα με τη χρήση κάποιου Dynamic Host Configuration Protocol (DHCP). Η διαδικασία αυτόματης ανάθεσης διεύθυνσης στο IPv6 περιλαμβάνει τη δημιουργία μιας τοπικής διεύθυνσης και η επαλήθευση της μοναδικότητάς της σε μια σύνδεση [16].
Το IPv6 ορίζει μηχανισμούς stateful και stateless address autoconfiguration.
Η stateless autoconfiguration δεν απαιτεί καμία χειροκίνητη διαμόρφωση των κόμβων, ελάχιστη (ή καθόλου) διαμόρφωση των δρομολογητών, και κανένα πρόσθετο server. Ο stateless μηχανισμός επιτρέπει σε έναν κόμβο να παράγει την διεύθυνσή του χρησιμοποιώντας έναν συνδυασμό τοπικά διαθέσιμων πληροφοριών και πληροφοριών που διαφημίζονται από τους δρομολογητές. Οι δρομολογητές διαφημίζουν prefixes που προσδιορίζουν το υποδίκτυο που σχετίζεται την σύνδεση, ενώ οι κόμβοι παράγουν ένα "interface identifier" που χαρακτηρίζει μοναδικά μια διεπαφή σε ένα υποδίκτυο. Μια διεύθυνση διαμορφώνεται με το συνδυασμό των δύο. Ελλείψει δρομολογητών, ένας κόμβος μπορεί μόνο να παραγάγει link-local διευθύνσεις, οι οποίες είναι ικανοποιητικές για την επικοινωνία μεταξύ των κόμβων του ίδιου link.
Στη stateful autoconfiguration, οι κόμβοι λαμβάνουν τις διευθύνσεις διεπαφών ή/και πληροφορίες διαμόρφωσης από έναν κεντρικό υπολογιστή (DHCPv6). Οι κεντρικοί υπολογιστές διατηρούν μια βάση δεδομένων που κρατάν τις ήδη χρησιμοποιούμενες διευθύνσεις. Οι δύο μηχανισμοί αλληλοσυμπληρώνονται. Παραδείγματος χάριν, ένας κόμβος μπορεί να χρησιμοποιήσει τον stateless μηχανισμό για να διαμορφώσει τις διευθύνσεις του, αλλά και τον stateful για να λάβει άλλες πληροφορίες.

Ο stateless μηχανισμός χρησιμοποιείται όταν δεν ενδιαφερόμαστε ιδιαίτερα για την ακριβή ανάθεση διευθύνσεων, εφόσον αυτές είναι μοναδικές και κατάλληλα δρομολογήσιμες. Η stateful προσέγγιση χρησιμοποιείται όταν απαιτείται αυστηρότερος έλεγχος στην ανάθεση των διευθύνσεων.

2.3.2.2.1. Duplicate Address Detection

Ας εξετάσουμε τον stateless μηχανισμό λεπτομερέστατα. Ο μηχανισμός αποτελείται από δύο διαδικασίες, την Duplicate Address Detection (DAD), και την Router Discovery. Αρχικά ο κόμβος αποδίδει στο interface την λεγόμενη *link-local* (τοπική δηλαδή για τη σύνδεση) διεύθυνση. Η διεύθυνση αυτή σχηματίζεται συνενώνοντας το well-known πρόθεμα της σύνδεσης FE80:0:0:0:0:0:0:0/64 [14] με το αναγνωριστικό (*identifier*) του interface, αντικαθιστώντας τα N τελευταία μηδενικά του προθέματος με τα N ψηφία του αναγνωριστικού. Τυπικά το N θα είναι 64 bits, και θα είναι στις περισσότερες περιπτώσεις η hardware διεύθυνση του κόμβου.
Πριν γίνει η απόδοση της link local διεύθυνσης στο interface ο κόμβος πρέπει να ελέγξει αν αυτή δεν χρησιμοποιείται από άλλον κόμβο στη σύνδεση (DAD). Ο έλεγχος αυτός γίνεται με τη χρήση ενός Neighbor Solicitation μηνύματος όπως αυτό ορίζεται από το NDP χρησιμοποιώντας ως αποδέκτη του μηνύματος την υποψήφια link-local διεύθυνση. Στην περίπτωση που υπάρχει κάποιος άλλος κόμβος στην σύνδεση με την ίδια link-local διεύθυνση θα απαντήσει με ένα Neighbor Advertisement μήνυμα, οπότε η διαδικασία διακόπτεται και το configuration του κόμβου πρέπει να συνεχιστεί χειρονακτικά. Για ευκολία υπάρχει η δυνατότητα να οριστεί και ένα εναλλακτικό αναγνωριστικό για το interface ώστε η διαδικασία να επαναληφθεί με το νέο αναγνωριστικό.
Αν δεν υπάρξει απάντηση σε ένα εύλογο χρονικό διάστημα ο κόμβος μπορεί να υποθέσει ότι η link-local διεύθυνση είναι μοναδική για τη σύνδεση και να προχωρήσει στην απόδοση αυτής στο interface, οπότε και ο κόμβος μπορεί να έχει επικοινωνία IP επιπέδου με τους υπόλοιπους κόμβους της σύνδεσης.

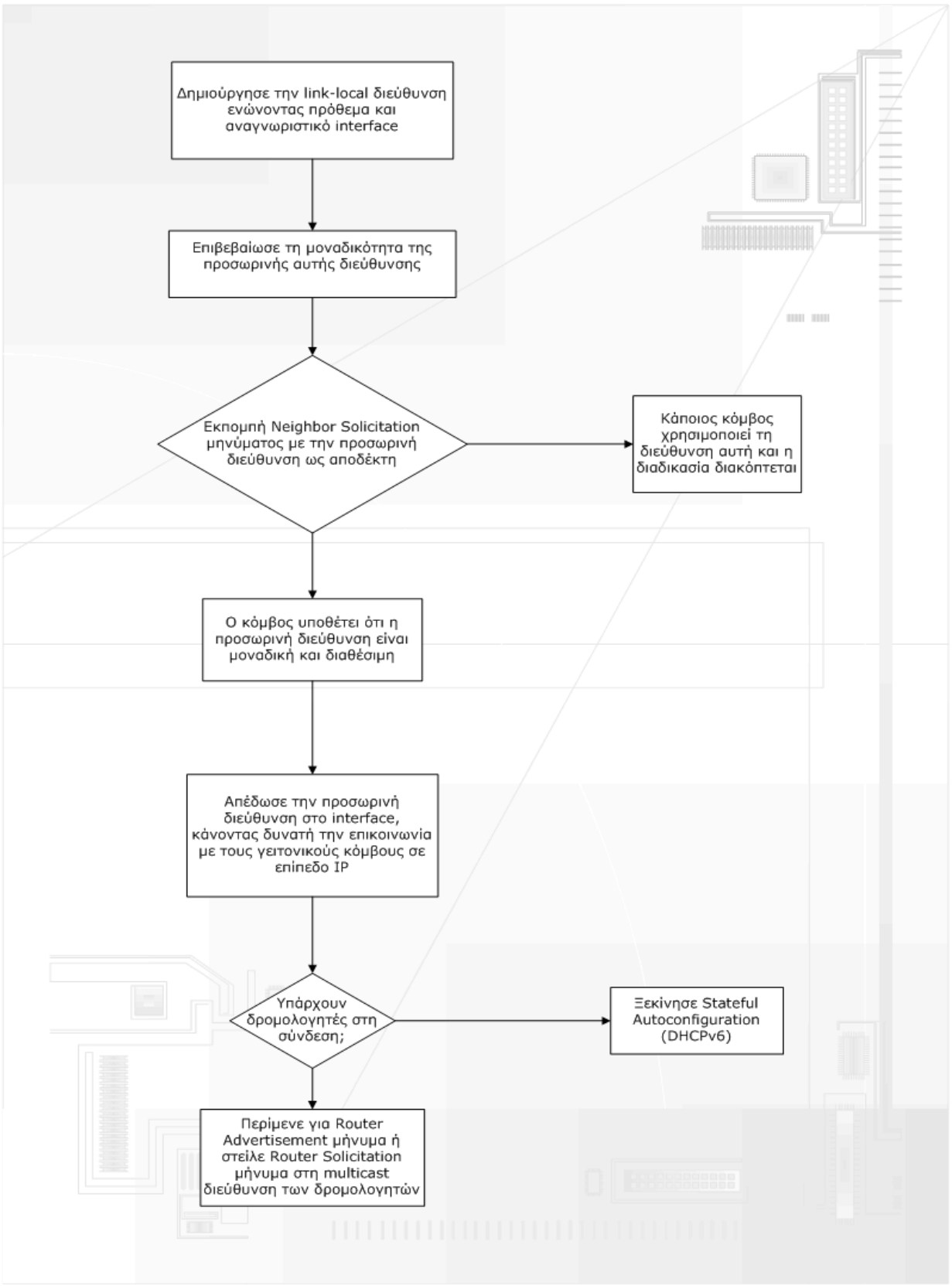
2.3.2.2.2. Router Discovery

Το επόμενο βήμα στον stateless μηχανισμό συνίσταται στο να ανιχνεύεται η παρουσία δρομολογητή (Router Discovery). Η λειτουργία του Router Discovery επιτρέπει οι μη-δρομολογητές να ζητήσουν και να επεξεργαστούν λαμβανόμενα Router Advertisements. Οι δρομολογητές στέλνουν unsolicited RA ανά ένα διάστημα μεταξύ του MIN_RTR_ADV_INT και του MAX_RTR_ADV_INT. Κάθε φορά που ένας router ανακαλύπτεται στο δίκτυο προστίθεται στην Default Routers List (DRL). Η DRL αποθηκεύει τους δρομολογητές που ο κόμβος μπορεί να στείλει off-link[8] πακέτα επίσης. Εάν δεν υπάρχει κανένας δρομολογητής στο δίκτυο τότε κάθε διεύθυνση θεωρείται on-link[9] [13]. Κατά την παραλαβή ενός RA από ένα κόμβο, τα προθέματα που διαφημίζονται από το δρομολογητή αναζητώνται και εάν το flag on-link έχει τεθεί για αυτό το πρόθεμα, τότε το πρόθεμα προστίθεται στην prefix list του κόμβου. Αυτά τα προθέματα συγκρίνονται με τις διευθύνσεις προορισμού όλων των πακέτων που στέλνονται. Αν οι διευθύνσεις ταιριάζουν με το πρόθεμα, τότε είναι on-link και άρα τα πακέτα μπορούν να σταλούν άμεσα σε έναν γειτονικό κόμβο χωρίς την επέμβαση του δρομολογητή.

Οι κατάλογοι DRL και on-link prefixes θεωρούνται Conceptual Data Structures (CDS) και χρησιμοποιούνται στον αλγόριθμο αποστολής. Είναι ένας αλγόριθμος που όλοι οι κόμβοι χρησιμοποιούν για να καθορίσουν πώς να διαβιβάσουν ή να στείλουν ένα πακέτο.

Παρακάτω παρουσιάζεται σχηματικά η Stateless Address Autoconfiguration τεχνική.

[8] Μία διεύθυνση που δεν έχει ανατεθεί σε κανένα interface στην συγκεκριμένη σύνδεση
[9] Μία διεύθυνση που έχει ανατεθεί σε κάποιο interface στην συγκεκριμένη σύνδεση



Εικόνα 10: Stateless Address Autoconfiguration

2.3.2.3. IPV6 Tunneling και Encapsulation

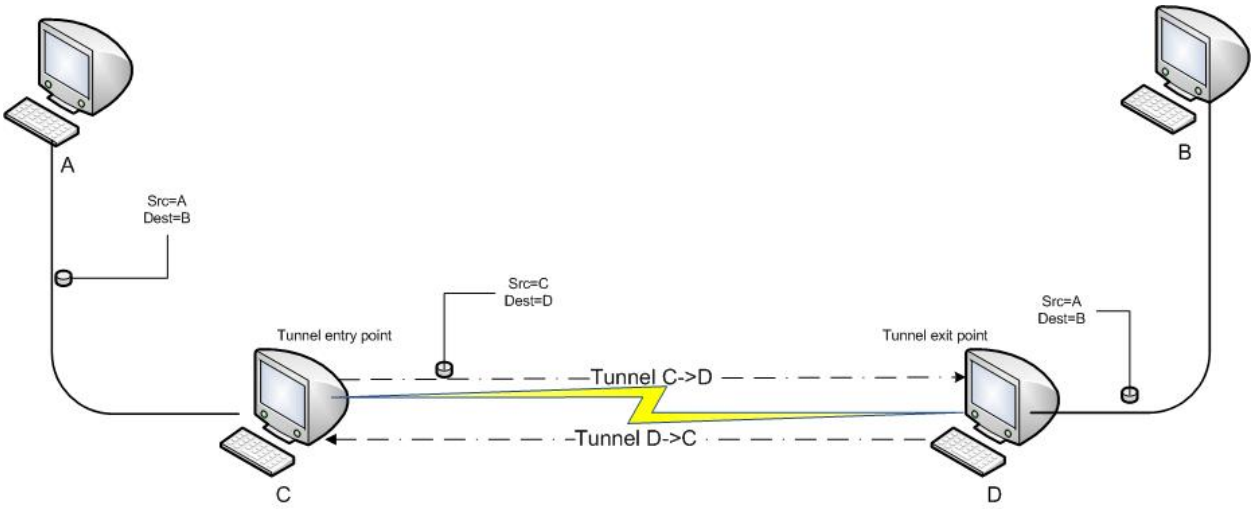
Το IPv6 tunneling [13] είναι μια τεχνική για την δημιουργία ενός ¨virtual link¨ μεταξύ δύο IPv6 κόμβων για τη διαβίβαση ολόκληρων πακέτων σαν περιεχόμενο άλλων πακέτων (Εικόνα 11). Από την άποψη των δύο κόμβων, αυτό το ¨virtual link¨, αποκαλούμενο IPv6 tunnel, δεν είναι τίποτα παραπάνω από μία point-to-point σύνδεση. Στην όλη διαδικασία παίρνουν μέρος τέσσερις κόμβοι. Οι A, B είναι η αντίστοιχοι αποστολέας και δέκτης, ενώ οι C και D είναι οι κόμβοι εισόδου και εξόδου του τούνελ (tunnel entry point, tunnel exit point).

Η IPv6 encapsulation (ενθυλάκωση) ορίζεται σαν η εισαγωγή σε ένα πακέτο μιας επιπλέον επικεφαλίδας, ή πολύ συχνά ένα σετ επικεφαλίδων επέκτασης (Εικόνα), οι οποίες καλούνται tunnel headers. Η ενθυλάκωση πραγματοποιείται σε ένα tunnel entry point, ως αποτέλεσμα της αποστολής του αρχικού πακέτου επάνω στο ¨virtual link¨. Το αρχικό πακέτο επεξεργάζεται κατά τη διάρκεια της διαβίβασης σύμφωνα με τους συγκεκριμένους κανόνες του πρωτοκόλλου του πακέτου. Παραδείγματος χάριν εάν το αρχικό πακέτο είναι:

- πακέτο IPv6, στην αρχική IPv6 επικεφαλίδα, το hop limit μειώνεται κατά ένα.
- πακέτο IPv4, στην αρχική IPv4 επικεφαλίδα, το πεδίο Time To Live (TTL) μειώνεται κατά ένα.

Σε μία λοιπόν αποστολή πακέτου από τον A, ο C λαμβάνει το πακέτο. Το ενθυλακώνει σε ένα νέο πακέτο με source address την διεύθυνσή του και destination address την διεύθυνση του D και το προωθεί στον D μέσω του τούνελ. Ο D με τη σειρά του εξάγει το αρχικό πακέτο και το προωθεί στον B, ο οποίος δεν γνωρίζει ότι το πακέτο πέρασε μέσα από τούνελ.

Ένα IPv6 τούνελ ένας κατευθυνόμενος μηχανισμός - η ροή πακέτων πραγματοποιείται μόνο προς μια κατεύθυνση μεταξύ των κόμβων εισόδου και εξόδου. Αμφίδρομη επικοινωνία επιτυγχάνεται με την χρήση δύο κατευθυνόμενων τούνελ όπως φαίνεται στην Εικόνα 11.



Εικόνα 11: IPv6 Encapsulation

2.3.3. Mobile IPv6

Το Mobile IPv6 [17] σχεδιάστηκε με βάση το MIP, χωρίς όμως τα μειονεκτήματα του. Έτσι περιληπτικά, το MIPv6 επιτρέπει σε έναν κινητό κόμβο να κινηθεί από μια σύνδεση προς άλλη χωρίς αλλαγή της Home Address του. Τα πακέτα μπορούν να δρομολογηθούν χρησιμοποιώντας αυτήν την διεύθυνση ανεξαρτήτως από το τρέχον σημείο σύνδεσης του κινητού κόμβου στο Διαδίκτυο. Ο κινητός κόμβος μπορεί επίσης να συνεχίσει να επικοινωνεί με άλλους κόμβους (στάσιμους ή κινητούς) μετά την κίνηση του σε ένα νέο link. Η μετακίνηση ενός κινητού κόμβου μακριά από το home network του είναι έτσι διαφανής για τα πρωτόκολλα υψηλότερου επιπέδου και τις εφαρμογές.

Το Mobile IPv6 είναι εξίσου κατάλληλο για κίνηση σε ομοιογενή δίκτυα, όπως και σε ετερογενή. Παραδείγματος χάριν, το MIPv6 διευκολύνει τη μετακίνηση κόμβων από ένα τμήμα Ethernet, σε ένα άλλο, καθώς επίσης και διευκολύνει τη μετακίνηση κόμβων από ένα τμήμα Ethernet σε ένα ασύρματο κύτταρο του τοπικού LAN, με τη διεύθυνση IP του κινητού κόμβου να παραμένει αμετάβλητη παρά τη μετακίνηση.

Ο σχεδιασμός του MIPv6 ευνοείται και από το ήδη υπάρχον MIPv4 [5], καθώς και από τις ευκαιρίες που παρέχει το IPv6. Συνεπώς το MIPv6 μοιράζεται πολλά χαρακτηριστικά από το MIPv4, αλλά ταυτόχρονα ενσωματώνεται στο IPv6

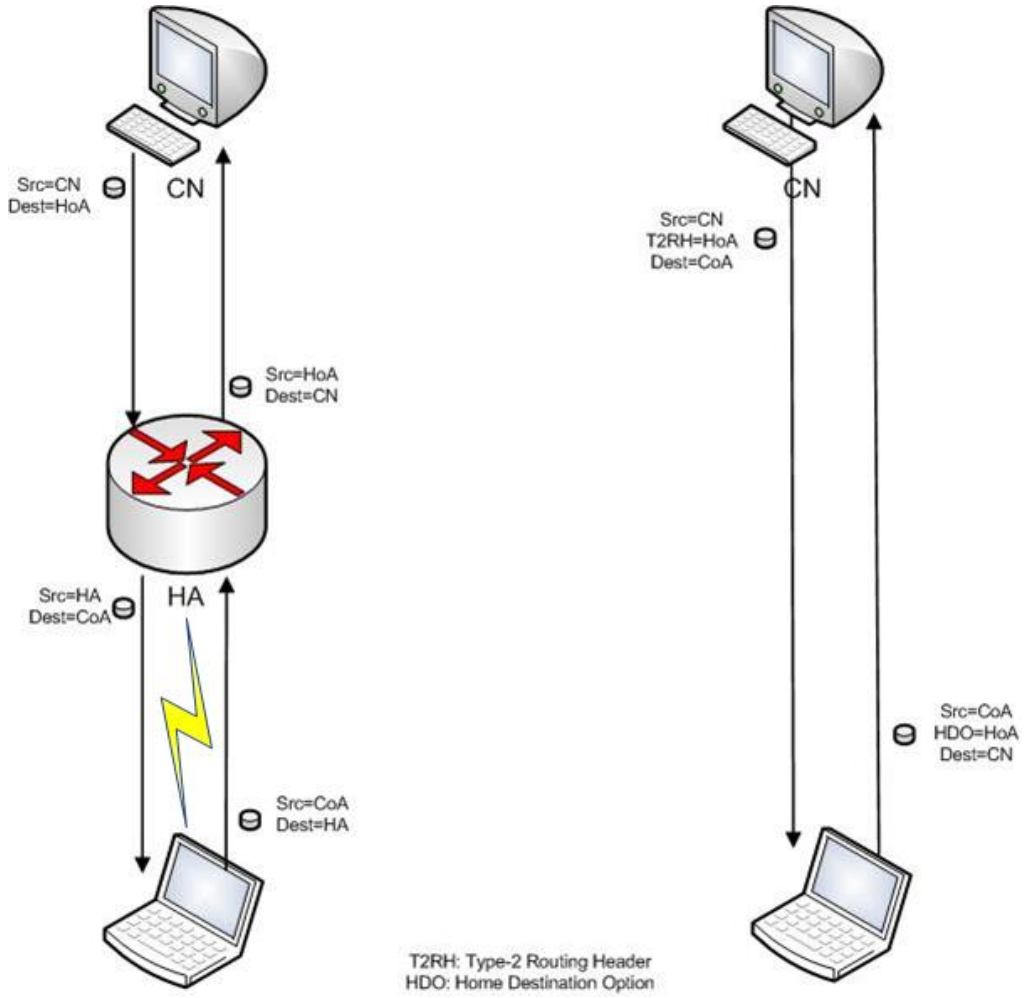
προσφέροντας έτσι πολλές άλλες βελτιώσεις. Συγκρίνοντας λοιπόν το Mobile IPv6 με το MIPv4 έχουμε:

- Δεν υπάρχει καμία ανάγκη χρήσης ειδικών δρομολογητών σαν ¨foreign agents¨, όπως στο MIPv4. Το MIPv6 λειτουργεί σε οποιαδήποτε θέση χωρίς καμία ειδική υποστήριξη από τους τοπικούς δρομολογητές.
- Το Route Optimization [17] είναι ένα θεμελιώδες μέρος του πρωτοκόλλου, παρά μία μη τυποποιημένη επέκταση [10].
- Το Route Optimization στο MIPv6 μπορεί να λειτουργήσει με ασφάλεια ακόμη και χωρίς την ύπαρξη προκαθορισμένων τροποποιήσεων.
- Στο MIPv6 υπάρχει ενσωματωμένη υποστήριξη χρήσης Route Optimization ακόμα και για δρομολογητές που εκτελούν ¨ingress filtering¨ [18].
- Η τεχνική IPv6 Neighbor Unreachability Detection επιβεβαιώνει συμμετρική προσπελασιμότητα μεταξύ του κινητού κόμβου και του δρομολογητή στην τρέχουσα θέση.
- Τα περισσότερα πακέτα που στέλνονται σε έναν κινητό κόμβο ενώ βρίσκεται μακριά από το home network του στέλνονται χρησιμοποιώντας μια IPv6 επικεφαλίδα, παρά με IP ενθυλάκωση, μειώνοντας έτσι το overhead σε σχέση με το MIPv4.
- Το MIPv6 δεν εξαρτάται από κανένα link-layer επίπεδο, δεδομένου ότι χρησιμοποιεί IPv6 Neighbor Discovery[13], αντί για ARP. Αυτό βελτιώνει επίσης την ευρωστία του πρωτοκόλλου.
- Η χρήση της IPv6 ενθυλάκωσης (και του Routing header) απαλείφει την ανάγκη διαχείρισης του ¨tunnel soft state¨[8].
- Ο μηχανισμός αυτόματης home agent address discovery επιστρέφει ένα μόνο reply στον κινητό κόμβο. Η broadcast προσέγγιση του IPv4 επιστρέφει χωριστά reply από κάθε home agent.

Η επικοινωνία στο MIPv6 γίνεται με έναν από τους δύο ακόλουθους τρόπους. Ο default τρόπος χρησιμοποιεί τούνελ μέσω του HA, ενώ ο προτιμημένος τρόπος είναι μια άμεση διαδρομή που καθιερώνεται μετά από Route Optimization. Και οι δύο τρόποι φαίνονται στην Εικόνα 12. Αντίθετα από το MIPv4, η τριγωνική δρομολόγηση δεν είναι πια μέθοδος επικοινωνίας αν και αυτό μπορεί εμφανιστείτε στιγμιαία κατά τη διάρκεια της φάσης μετάβασης μεταξύ των δύο αναφερθέντων τρόπων.

2.3.3.1. MIPV6 Δρομολόγηση με χρήση Tunneling

Όσο ο MN βρίσκεται μακριά από το σπίτι ο HA λειτουργεί σαν proxy. Αυτό σημαίνει πως οποιαδήποτε πακέτα απευθύνονται στον MN θα καταλήξουν στον HA, καθώς αυτός θα ανταποκριθεί σε όλα Neighbor Solicitation requests για τον MN.
Μόλις ο HA παραλάβει ένα πακέτο θα το προωθήσει στον MN στην τρέχουσα θέση του μέσω της CoA που βρει στην binding cache του. Η εγγραφές στην binding cache του δημιουργούνται όταν ο MN εγγράφηκε στον HA και ανανεώνονται με κάθε Binding Update (BU) από τον MN. Όπως και στο κλασσικό IPv6 ο HA θα ενθυλακώσει το αρχικό πακέτο σε ένα νέο. Η tunnel επικεφαλίδα θα έχει μια διεύθυνση προέλευσης την IP διεύθυνση του HA και διεύθυνση προορισμού την CoA διεύθυνση του MN. Ο MN απομονώνει το αρχικό πακέτο, το οποίο πια φαίνεται λες και ο CN το είχε στείλει απευθείας στον MN.
Στην περίπτωση που ο MN δεν έχει δημιουργήσει binding με τον CN, θα πρέπει να στείλει όλα τα πακέτα που προορίζονται για τον CN μέσω του HA χρησιμοποιώντας reverse tunneling. Το αρχικό πακέτο έχει διεύθυνση προέλευσης την HoA και διεύθυνση προορισμού τον CN, ενώ η tunneling επικεφαλίδα θα έχει διεύθυνση προέλευσης την CoA του MN και προορισμό την διεύθυνση του HA. Μόλις ο HA λάβει το πακέτο θα ελέγξει αν η διεύθυνση προέλευσης της tunneling επικεφαλίδας είναι η CoA που αντιστοιχεί στην HoA του αρχικού πακέτου, εμποδίζοντας έτσι άλλους κόμβους να μεταμφιέζονται σαν MN. Κατά συνέπεια όταν το πακέτο φτάσει στον CN μοιάζει σαν ο MN να το είχε στείλει από τον home network του.



Εικόνα 12: Τρόποι επικοινωνίας στο MIPv6

2.3.3.2. MIPV6 Δρομολόγηση με χρήση Route Optimization

Αυτός ο τρόπος παράδοσης πακέτων δεν απαιτεί τη μεσολάβηση του HA, και συνεπώς επιτρέπει γρηγορότερη και πιο αξιόπιστη μετάδοση. Αυτό επιτυγχάνεται με χρήση του πεδίου *home address destination* και της type-2 επικεφαλίδας [17]. Η χρήση αυτών των δύο εξομοιώνει τους μηχανισμούς ενθυλάκωσης της προηγούμενης μεθόδου, αλλά επιφέρει ελάχιστο overhead. Το πεδίο *home address destination* του MN περιέχει τη HoA. Αυτό επιτρέπει σε ένα κινητό κόμβο να στείλει πακέτα με διεύθυνση προέλευσης την CoA, πράγμα που είναι τοπολογικά ορθό, και

συνεπώς περνάει τους ingress filtering κανόνες του ξένου δρομολογητή. Όταν το πακέτο φτάσει, ο CN θα αντιστρέψει το *home address destination* με την διεύθυνση προέλευσης του πακέτου. Το τροποποιημένο πακέτο μεταφέρεται στο transport layer και έτσι η εφαρμογή δεν αντιλαμβάνεται καν ότι επικοινωνεί με ένα κινητό κόμβο.
Μια παρόμοια διαδικασία εμφανίζεται και όταν ο CN στέλνει δεδομένα στον MN. Η εφαρμογή απευθύνει το πακέτο στην HoA του MN. Στο network layer ο CN θα ελέγξει την binding cache του προκειμένου να ανακαλύψει την τρέχουσα θέση του MN, δηλαδή την CoA που ανέφερε ο MN με το BU του. Θα προσθέσει μια type-2 επικεφαλίδα στον πακέτο και θα αντικαταστήσει τη διεύθυνση προέλευσης με την CoA. Το πακέτο θα ταξιδέψει μέσω του δικτύου χρησιμοποιώντας κανονικές διαδικασίες και φθάνει στον MN. Ο MN θα επεξεργαστεί την type-2 επικεφαλίδα ανταλλάσσοντας τα περιεχόμενα του με τη διεύθυνση προέλευσης του πακέτου. Κατά συνέπεια το τελικό πακέτο που περνά στο transport layer έχει ως διεύθυνση προέλευσης την HoA. Αυτό κρατά τις εφαρμογές ανίδεες της μετακίνησης του κόμβου.
Για να καθιερωθεί μια άμεση διαδρομή, ο MN πρέπει να στέλνει BU με την τρέχουσα CoA του στον CN, ο οποίος την αποθηκεύει στην bonding cache του. Προκειμένου να αποτραπεί από κακόβουλους κόμβους να μεταμφιέζονται σαν MNs στέλνοντας BUs με την HoA του MN, χρησιμοποιείται η διαδικασία *return routability* για να ελεγχθεί η αυθεντικότητα των κόμβων. Παρακάτω εξηγούμε πως αυτό γίνεται [17]. Σαν πρώτο βήμα ο MN στέλνει ένα *Home Test Init* μήνυμα στον CN για να αρχίσει τη διαδικασία *return routability*. Ο CN τότε θα στείλει ένα πακέτο δοκιμής σε κάθε μια από τις δύο διαφορετικές διαδρομές, μια χρησιμοποιώντας την HoA σαν προορισμό και μία χρησιμοποιώντας την CoA σαν προορισμό. Τα δύο πακέτα δοκιμής περιέχουν τα μέρη ενός *time cookie* που συναρμολογούνται στον MN και στέλνονται πίσω στον MN. Μόνο αν και οι δύο διευθύνσεις δείχνουν στον ίδιο κόμβο, θα μπορεί να λάβει ολόκληρο το time cookie. Αυτό βασίζεται στην υπόθεση ότι ο HA έχει πιστοποιήσει την ταυτότητα του MN. Αυτή είναι μια έγκυρη υπόθεση καθώς το MIPv6 έχει υιοθετήσει την χρήση της IPSec πιστοποίησης στα BU του MN στον HA [19, 20]. Κατά συνέπεια ο μηχανισμός του Return Routability θα προσθέσει ενάμισι round trip ανά CN για τον οποίο η διαδρομή θα βελτιστοποιηθεί.

Όσο αναφορά τον χειρισμό real-time κυκλοφορίας στο MIPv6, έχει εμφανώς καλύτερη συμπεριφορά καθώς αφενός δεν γίνεται τριγωνική δρομολόγηση καθόλου, και αφετέρου μπορεί πάντα να εφαρμοστεί Route Optimization εκτός και αν ο CN το έχει απαγορέψει. Και οι δύο τρόποι δρομολόγησης φαίνονται στην Εικόνα 12.

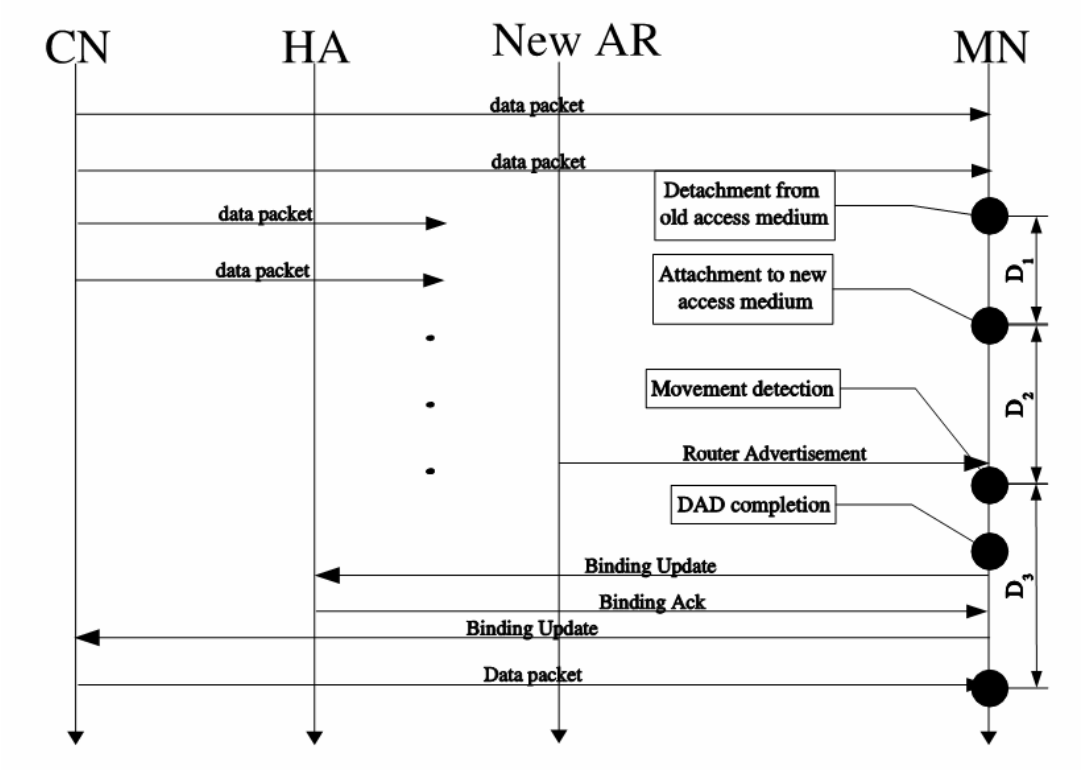
3. LOCAL AR MIPV6 HANDOVER EXTENSIONS

Όπως αναφέραμε στο κεφάλαιο 1 σαν handover καθυστέρηση ορίζουμε τον χρόνο που μεσολαβεί από τη στιγμή που ο MN χάνει την σύνδεση του με το παλιό μέσο πρόσβασης, έως τη στιγμή που ανακτά επικοινωνία με χρήση του νέου μέσου πρόσβασης. Στην Εικόνα 13 παρουσιάζεται η τυπική διαδικασία handover που ακολουθείτε στο κλασσικό MIPv6 [17]. Υπάρχουν τρία στοιχεία που καθορίζουν το χρονικό πλάτους του handover. Το πρώτο, D_1, αφορά την καθυστέρηση λόγω του L2 handover, πχ η μετάβαση σε ένα 802.11b δίκτυο από ένα access point σε ένα άλλο. Το D_2 είναι ο χρόνος που χρειάζεται ο MN να αντιληφθεί την ύπαρξη ενός νέου AR και να δημιουργήσει μια νέα CoA. Αυτός ο χρόνος εξαρτάται από το μέγεθος της επικάλυψης ή της απόστασης δύο AP, την ταχύτητα του κινητού κόμβου και τον ρυθμό των unsolicited Router Advertisements. Ο D_2 καλείται και χρόνος ραντεβού (rendezvous time). Το τρίτο στοιχείο, D_3, είναι ο χρόνος που απαιτείται για την αποστολή BU στους HA και CN, συν τον χρόνο για την συνέχιση της επικοινωνίας, δηλαδή τον χρόνο που χρειάζεται ένα νέο πακέτο να φτάσει στον MN από τον νέο AR. Ο D_3 καλείται επίσης και χρόνος καταχώρησης (registration time).

Τα L2 handovers εξαρτούνται αποκλειστικά από τις συγκεκριμένες τεχνολογίες μεταφοράς που χρησιμοποιούνται και συνεπώς τίποτα δεν μπορεί να γίνει για να αλλάξουμε την συμπεριφορά τους. Για αυτό ακριβώς το λόγο δεν θα ασχοληθούμε περαιτέρω με τα L2 handovers.

Υπάρχουν βασικά δύο τεχνικές διαχείρισης των handovers, οι προφητικές και οι μη-προφητικές. Οι προφητικές απαιτούν βοήθεια από την υποδομή του δικτύου, λειτουργία που συνήθως δεν ενσωματώνεται στο σύστημα. Είναι προφητική επειδή προβλέπει σε ποια ασύρματη σύνδεση και ως εκ τούτου σε ποιο υποδίκτυο θα κινηθεί ο MN πριν το πραγματικό handover γίνει. Οι μη-προφητικές μέθοδοι γενικά δεν απαιτούν καμία ειδική βοήθεια από την υποδομή και είναι αντιδραστικής φύσης, δηλαδή ένας MN θα εκτελέσει L3 handover μόνο αφού έχει ήδη ανιχνεύσει μια μετάβαση σε ένα διαφορετικό L3 υποδίκτυο. Κατά γενικό κανόνα οι προφητικές μέθοδοι είναι πιο σύνθετες στην εφαρμογή και στη διαχείριση.

Παρακάτω περιγράφουμε κάποιες επεκτάσεις που έχουν προταθεί για την μείωση της handover καθυστέρησης.



Εικόνα 13: Mobile IPv6 Handover

3.1. Fast Handovers for Mobile IPv6

Ένα παράδειγμα μια προφητικής τεχνικής είναι η Fast Handovers for MIPv6 [21]. Το πρωτόκολλο επιτρέπει σε έναν MN να ανιχνεύσει γρήγορα ότι έχει κινηθεί προς ένα νέο υποδίκτυο, γνωρίζοντας το νεό access point και το σχετιζόμενο subnet prefix του, ενώ ο MN είναι ακόμα συνδεδεμένος με το οικίο δίκτυο του. Παραδείγματος χάριν, ένας MN μπορεί να ανιχνεύσει διαθέσιμα AP χρησιμοποιώντας link-layer μηχανισμούς (πχ "scan" σε WLAN) και να ζητήσει τις πληροφορίες υποδικτύου που αντιστοιχούν σε ένα ή περισσότερα από τα ανακαλυμμένα σημεία πρόσβασης. Το αποτέλεσμα είναι η δυάδα [AP-ID, AR-info] μέσω των οποίων ο MN μπορεί εύκολα να ανιχνεύσει μετακίνηση του. Έτσι αν συνδεθεί σε ένα σημείο πρόσβασης με AP-ID θα γνωρίζει επιπλέον το prefix του νέου router, την IP διεύθυνση, και την L2

διεύθυνση. Τα *Router Solicitation for Proxy Advertisement* (RtSolPr) και *Proxy Router Advertisement* (PrRtAdv) μηνύματα βοηθούν στην ανίχνευσης της μετακίνησης.

Στην Εικόνα 14 παρουσιάζονται τα βήματα της διαδικασίας για fast handovers. Ο MN ξέρει ότι πρόκειται να κάνει handover, ίσως λόγω ασθενούς σήματος από τον PAR. Έτσι ο MN στέλνει ένα RtSolPr στον τρέχον AR (PAR). Ο PAR απαντά με ένα PrRtAdv που περιέχει όλα τα χαρακτηριστικά του NAR που περιγράψαμε παραπάνω. Με αυτά ο MN διαμορφώνει μία νέα CoA ενώ είναι ακόμα συνδεδεμένος με τον PAR. Ως εκ τούτου, η καθυστέρηση ανακάλυψης προθέματος μηδενίζεται.

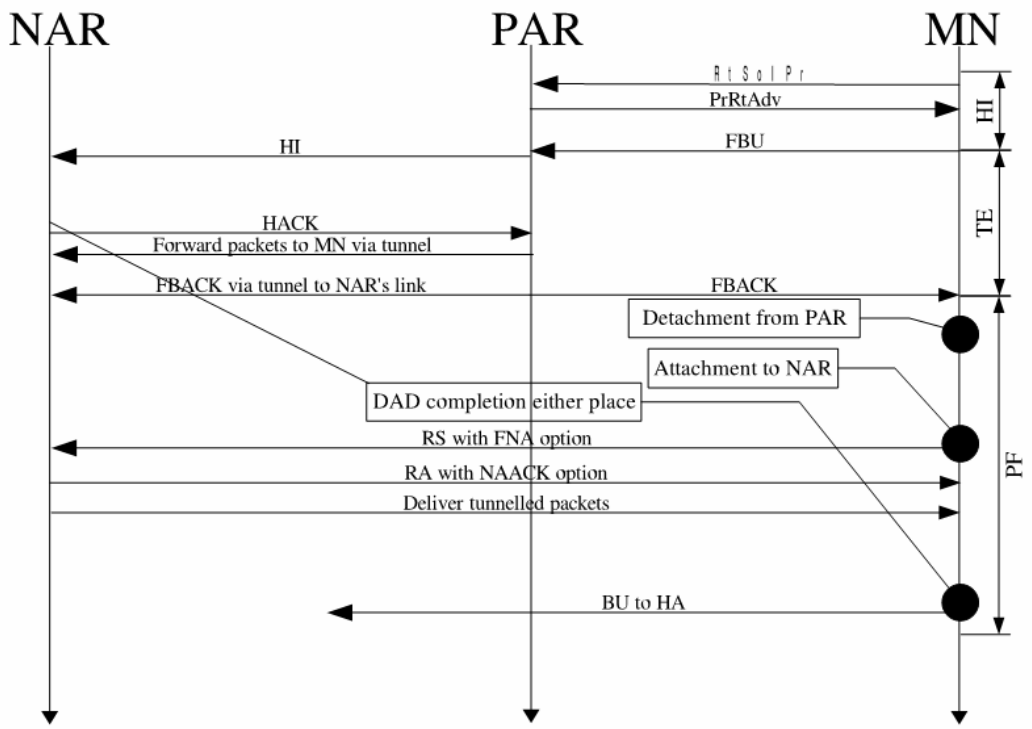
Σε αυτό το σημείο ο MN στέλνει ένα *Fast Binding Update* (FBU). Μόλις ο PAR λάβει το FBU, θα στείλει ένα *Handover Initiate* (HI) στον NAR. Σκοπός του HI είναι η δημιουργία ενός αμφίδρομου τούνελ μεταξύ PAR και NAR με στόχο τη χρήση της PCoA στο δίκτυο του NAR, όπως και την εξασφάλιση της μοναδικότητας της NCoA. Ο NAR με τη σειρά του θα επιστρέψει ένα *Handover Acknowledgment* (HACK) μήνυμα στον PAR, ο οποίος θα στείλει *Fast Binding Acknowledgement* (FBack) στον MN. Από αυτό το σημείο ο MN μπορεί να χρησιμοποιεί την NCoA.

Μόλις ο MN συνδεθεί με το νέο link, θα στείλει RS στον NAR, στο οποίο θα ενσωματώσει ένα *Fast Neighbor Advertisement* (FNA) μήνυμα. Με το FNA μήνυμα θα αρχίσει η προώθηση όσων πακέτων απευθύνονται στην PCoA στον νέο σύνδεσμο, δημιουργώντας μια εγγραφή στην binding cache του NAR. Εάν το FBack έδειξε πως η NCoA ήταν αποδεκτή τότε ο MN μπορεί να την χρησιμοποιήσει σαν source address αφού στείλει BU στον CN και στον HA.

Στην περίπτωση που το FBack έδειξε πως η NCoA δεν ήταν αποδεκτή για οποιοδήποτε λόγο, τότε η ένδειξη *Neighbor Advertisement Acknowledgement* (NAACK) συμπεριλαμβάνεται στα RA του NAR, προτείνοντας ίσως μια διαθέσιμη NCoA. Το FMIPv6 επιτρέπει στον MN να χρησιμοποιεί την PCoA για ένα μικρό διάστημα έως ότου λάβει μια έγκυρη NCoA για τις εξερχόμενες επικοινωνίες του, μειώνοντας έτσι την απώλεια πακέτων.

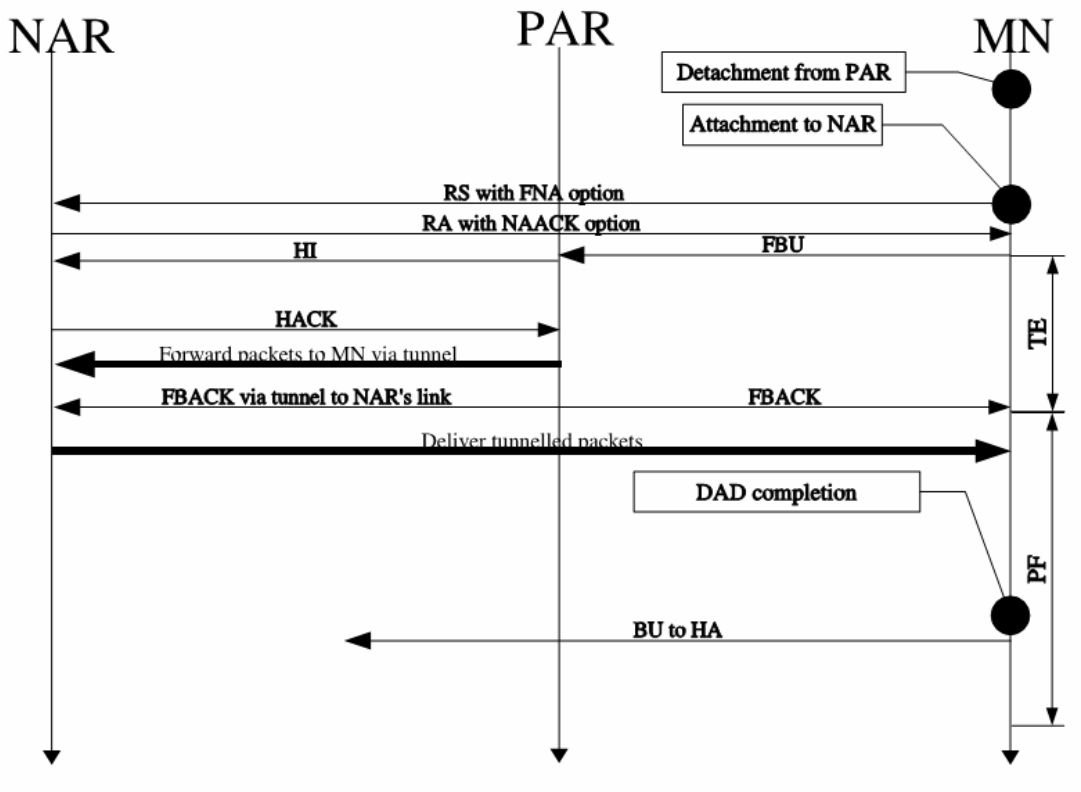
Όταν δεν είναι διαθέσιμη η a priori ενημέρωση για τα διαθέσιμα AP, ή όταν τα AP δεν είναι επικαλυπτόμενα χρησιμοποιείται μια διαφορετική τεχνική που φαίνεται στην Εικόνα 15. Η συγκεκριμένη τεχνική δεν απαιτεί γνώση του NAR πριν την σύνδεση. Δεν υπάρχει καμία ανταλλαγή PtRtAdv ή RtSolPr μηνυμάτων, και γενικά η φάση του

Handover Initiation δεν συμβαίνει καθόλου. Μόλις ο MN συνδεθεί με τον NAR θα στείλει ένα RS με FNA. Αυτό είναι αναγκαίο προκείμενου ο MN να γνωρίσει την Ip και link-layer διεύθυνση του δρομολογητή. Κατόπιν στέλνει ένα FBU στον PAR χρησιμοποιώντας την PCoA ως διεύθυνση προέλευσης.



Εικόνα 14: Anticipated Fast Handover

Ο NAR εάν υποστηρίζει FMIPv6 πρέπει να επιτρέψει στους ingress filtering κανόνες οποιαδήποτε πακέτα προορίζονται στον PAR με διεύθυνση προέλευσης που ανήκει στο υποδίκτυο του PAR. Μόλις το FBU φτάσει στον PAR, ο PAR θα πρέπει να ξεκινήσει τις διαδικασίες δημιουργίας τούνελ. (HI και HACK μηνύματα). Στο μεταξύ ο MN μπορεί να χρησιμοποιήσει την PCoA καθώς το αμφίδρομο τούνελ πρέπει να έχει δημιουργηθεί σε αυτό το στάδιο. Η λήψη του FBack αποτελεί επιβεβαίωση λειτουργίας του τούνελ. Ο MN είναι πια ελεύθερος να διαμορφώσει NCoA μόλις λάβει RA, το οποίο πρέπει να περιλάβει ένα NAACK που θα δείχνει πως η NCoA δεν είναι έγκυρη και να αναγκάσει εον MN να διαμορφώσει την νέα του διεύθυνση με τις τεχνικές που περιγράφονται στην παράγραφο 2.3.2.2.



Εικόνα 15: Non-anticipated Fast Handover

Αυτή η τεχνική δεν είναι ικανή να μηδενίσει την rendezvous καθυστέρηση όπως η πρώτη, μπορεί όμως να μειώσει την ολική καθυστέρηση καθώς τα πακέτα διαβιβάζονται γρήγορα από τον PAR στον NAR, επιτρέποντας στον MN να συνεχίσει την επικοινωνία του μέσω του τούνελ, χωρίς να περιμένει την ολοκλήρωση της εγγραφής του με τον HA όπως το βασικό MIPv6 θα απαιτούσε.

3.2. Layer 2 Triggers for Mobile IPv6

Στην αρχή του Κεφαλαίου 3 είδαμε πως ο rendezvous time εξαρτάται εκτός των άλλων από τον χρόνο που χρειάζεται ο MN να αντιληφθεί ότι έχει μεταβεί σε ένα νέο δίκτυο. Επίσης στην παράγραφο 2.1.3 παρουσιάσαμε τα L2 και L3 handovers και είδαμε πως ένα L3 handover έπεται πάντα ενός L2 handover. Συνδυάζοντας αυτά τα δύο και προσπερνώντας την μάλλον ακριβή υλοποίηση FMIPv6 οδηγούμαστε στην τεχνική των L2 triggers. Το μόνο που προϋποθέτει αυτή η τεχνική είναι την

ενημέρωση του network layer από το link layer κάθε φορά που συμβαίνει ένα L2 handover. Κάτι τέτοιο δεν απαιτεί καμία αλλαγή στην υποδομή του δικτύου και συνεπώς είναι μια απλή και φτηνή επιλογή. Αυτή λοιπόν η απλή ανακοίνωση μπορεί να βοηθήσει στη μείωση του rendezvous χρόνου, αφού ωθεί τον MN να ξεκινήσει πιο γρήγορα την handover διαδικασία. Όπως ήδη αναφέραμε ένα L2 handover δεν συνδέεται πάντα με ένα L3 handover, και συνεπώς υπάρχει περίπτωση ο L2 trigger να είναι λανθασμένος. Κάτι τέτοιο δεν αποτελεί σοβαρό πρόβλημα αφού το μόνο που θα στοιχίσει στο δίκτυο είναι ένα RS και το αντίστοιχο RA που θα ενημερώνει τον MN ότι δεν έχει αλλάξει L3 network και δεν πρέπει έτσι να αρχίσει το L3 handover. Έτσι τα οφέλη των L2 triggers αντισταθμίζουν κατά πολύ το κόστος, ακόμα και στις περιπτώσεις που τα L2 handovers είναι πολύ συχνά και καταλαμβάνουν αξιοσημείωτο bandwidth, καθώς η αύξηση των AP θα έλυνε κάποιο τέτοιο ενδεχόμενο.

3.3. Fast Solicited Router Advertisements

Σύμφωνα με το [13] ένας δρομολογητής πρέπει να καθυστερεί την απάντηση σε ένα RS για έναν τυχαίο χρόνο μεταξύ 0 και MAX_RA_DELAY_TIME[10] δευτερολέπτων. Ο λόγος είναι ότι εάν υπάρχουν περισσότεροι από ένας δρομολογητές στη σύνδεση, και όλοι απαντήσουν αμέσως στο RS, τότε τα RA θα συγκρουστούν. Ένας άλλος λόγος που συμβαίνει αυτό είναι για να αποφευχθεί η συμφόρηση του συνδέσμου όταν όλοι οι routers απαντούν.

Ο αντίκτυπος αυτού του περιορισμού μπορεί να είναι πολύ σοβαρός. Εξετάστε για παράδειγμα την περίπτωση που ένας κόμβος δέχεται ένα L2 trigger, όπως περιγράψαμε παραπάνω. Θα μπορούσε αμέσως να στείλει ένα RS από το να περιμένει τα περιοδικά RA των δρομολογητών. Εντούτοις, αν ο δρομολογητής συμφωνεί με το [13], τότε πρέπει να περιμένει προτού απαντήσει αυξάνοντας το χρονικό διάστημα εγκαθίδρυσης της σύνδεσης.

[10] MAX_RA_DELAY_TIME 500ms default

Για να επιτραπούν γρηγορότεροι χρόνοι απόκρισης στην επεξεργασία των RS, πρέπει να επιτρέψουμε το πολύ σε ένα δρομολογητή σε οποιαδήποτε σύνδεση να αποκρίνεται αμέσως στα RS. Ένα RA που αποστέλλεται αμέσως στον αποστολέα και όχι καθυστερημένα είναι γνωστό ως *Fast RA* [22].

Ένας δρομολογητής που είναι ορισμένος σαν *Fast RA router* διατηρεί έναν μετρητή, FastRACounter, που μετρά το τελευταίο Fast RA που έστειλε από το τελευταίο solicited multicast RA. Όταν παραλαμβάνεται ένα RS, ένα RA πρέπει να σταλεί αμέσως εάν:

$$\text{FastRACounter} \leq \text{MAX\_FAST\_RAS},$$

όπου MAX_FAST_RAS είναι ο μέγιστος επιτρεπόμενος αριθμός σταλθέντων Fast RA πριν από ένα multicast RA.

Κάθε φορά που στέλνεται ένα Fast RA, ο FASTRACounter πρέπει να αυξηθεί κατά ένα. Τυπικά το MAX_FAST_RAS είναι 10, αλλά πρέπει να διαμορφώνεται σύμφωνα με την χωρητικότητα των δρομολογητών και το αναμενόμενο φορτίο RS. Όταν το FastRACounter υπερβαίνει το MAX_FAST_RAS ένα multicast RA πρέπει να δρομολογηθεί όσο το δυνατόν πιο γρήγορα, βασισμένο στον περιορισμό ότι δύο διαδοχικά multicast RA πρέπει να απέχουν τουλάχιστον MIN_DELAY_BETWEEN_RAS[11] δευτερόλεπτα. Περαιτέρω BS πριν την αποστολή multicast RA απορρίπτονται. Το FastRACounter μηδενίζεται μετά από κάθε multicast PA.

3.4. Fast RA beacons

Η τεχνική των Fast RA beacons είναι μια ιδέα που βασίζεται στο βασικό MIPv6 και δεν απαιτεί καμία επέκταση. Όπως αναφέραμε και παραπάνω διαδοχικά multicast RA πρέπει να απέχουν τουλάχιστον MIN_DELAY_BETWEEN_RAS δευτερόλεπτα. Συνεπώς μια ιδέα για την μείωση του rendezvous χρόνου είναι να ωθήσουμε τους δρομολογητές να στέλνουν RA πιο συχνά, και όχι κάθε 3 δευτερόλεπτα. Έτσι χρησιμοποιώντας τις ελάχιστες τιμές στις μεταβλητές MinRtrAdvInterval και

[11] Ορίζεται στο [13] σαν 3 δευτερολεπτα

MaxRtrAdvInterval[12] - που υπερισχύουν της MIN_DELAY_BETWEEN_RAS – που μας επιτρέπει το [17] οι δρομολογητές στέλνουν πιο συχνά multicast RA επιτρέποντας στους κινητούς κόμβους να επιταχύνουν την handover διαδικασία.

3.5. Optimistic Duplicate Address Detection

Η τεχνική Optimistic Duplicate Address Detection (ODAD) [23] είναι μια τροποποίηση των διαδικασιών των [13, 16]. Σκοπός είναι η ελαχιστοποίηση της καθυστέρησης διαμόρφωσης διευθύνσεων στην επιτυχή περίπτωση. Η μέθοδος επιτρέπει στους κόμβους να έχουν και να χρησιμοποιούν μία *tentative* διεύθυνση, δηλαδή μια διεύθυνση της οποίας η μοναδικότητα δεν έχει επιβεβαιωθεί ακόμα. Αυτό δεν δημιουργεί πρόβλημα αν δεχτούμε πως οι διευθύνσεις είναι ομοιόμορφα κατανεμημένες σε όλο το διαθέσιμο φάσμα και συνεπώς η διαδικασία DAD σχεδόν πάντα πετυχαίνει[13]. Σαν αποτέλεσμα κόμβοι που υπόκεινται σε ODAD είναι ικανοί να συνεχίζουν τις επικοινωνίες τους πολύ νωρίτερα σε σχέση με ένα handover χωρίς ODAD.

Προκειμένου να αποφευχθούν παρεμβολές στην περίπτωση που η διεύθυνση ανήκει σε κάποιον άλλο κόμβο, είναι σημαντικό ότι ένας MN που υπόκειται σε ODAD να μην στέλνει μηνύματα από μια αισιόδοξη διεύθυνση, καθώς κάτι τέτοιο θα επηρέαζε τις neighbor caches των γειτόνων του. Αυτό επιτυγχάνεται:

- Καθαρίζοντας το flag *Override* στα RA που προορίζονται για σε αισιόδοξες διευθύνσεις, αποτρέποντας έτσι τους γείτονες να αλλάξουν εγγραφές στις Neighbor caches τους με λανθασμένα στοιχεία. Αυτό επιτρέπει επίσης στον νόμιμο κόμβο να στείλει *Neighbor Advertisement* (NA) με το flag *Override* ενεργοποιημένο υπερασπίζοντας την διεύθυνση του και ενημερώνοντας παράλληλα τον MN ότι η διεύθυνση του χρησιμοποιείται ήδη και να

[12] MinRtrAdvInterval 0.03 seconds
MaxRtrAdvInterval 0.07 seconds

[13] Εάν ένας κόμβος κινείται μεταξύ δικτύων 50000 κόμβων κάθε ένα λεπτό για 100 χρόνια, η πιθανότητα σύγκρουσης της διεύθυνσης του είναι $1.3e10^{-6}$, δηλαδή μικρότερη από μία στο εκατομμύριο.

σταματήσει να την χρησιμοποιεί. Το flag *Override* καθορίζεται στο [13] και χρησιμοποιείται για Proxy Neighbor Advertisements.

- Απαγορεύοντας στον MN να στέλνει NS από μια αισιόδοξη διεύθυνση. Τα NS περιλαμβάνουν το πεδίο *source link-layer address* το οποίο μπορεί να προκαλέσει λάθη στην Neighbor cache. Τα NS που στέλνονται από αισιόδοξους κόμβους έχουν άδειο το *source link-layer address* πεδίο.
- Απαγορεύοντας την χρησιμοποίηση μιας αισιόδοξης διεύθυνσης σαν *source link-layer address* στα RS. Τα RS στέλνονται χωρίς *source link-layer address*.
- Ο MN προωθεί πακέτα μόνο μέσω του δρομολογητή στον οποίο κάνει ODAD. Συνεπώς μόνο ο δρομολογητής γνωρίζει την ύπαρξη του κόμβου.

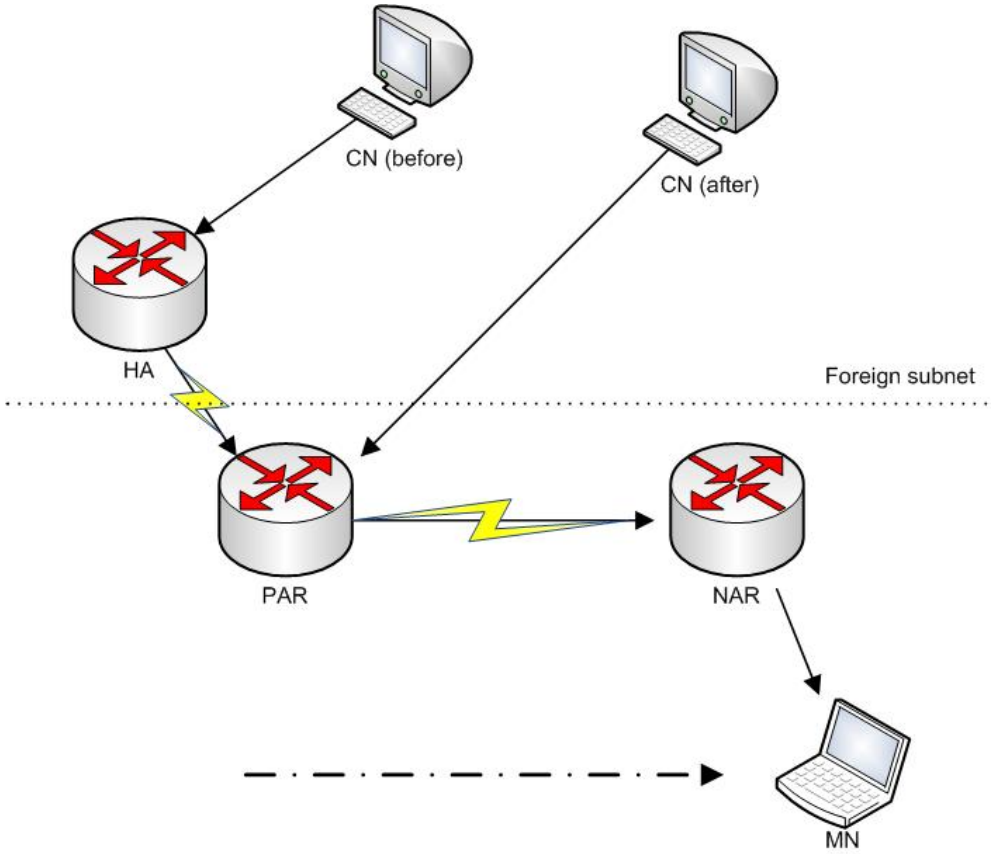
Υπάρχει επίσης μια επιπλέον διαδικασία που παράγει ένα νέο τυχαίο suffix και ως εκ τούτου ο MN μπορεί να διαμορφώσει μια άλλη IP διεύθυνση για να υποβληθεί στο DAD εκ νέου στην περίπτωση που το ODAD αποτύχει.

3.6. Previous Care-of-Address Forwarding

Η επέκταση Previous Care-of-Address Forwarding [24] περιγράφτηκε στις αρχικές εκδόσεις του MIPv6, αλλά τελικά απορρίφθηκε για λόγους ασφάλειας.

Σύμφωνα με αυτό όταν ένας κινητός κόμβος συνδέεται σε ένα νέο link και διαμορφώνει μια νέα CoA, μπορεί να προωθεί τα πακέτα του από την PCoA στην NCoA (Εικόνα 16). Για να κάνει κάτι τέτοιο ο MN στέλνει ένα BU στον agent της PCoA, υποδεικνύοντας την PCoA σαν HoA και την NCoA σαν CoA, έτσι ώστε να κάνει binding μεταξύ τους. Αυτό επιτρέπει σε κόμβους που δεν γνωρίζουν ακόμα την μετακίνηση του MN να του διαβιβάσουν πακέτα μέσω της PCoA, μειώνοντας έτσι την απώλεια πακέτων λόγο της μη ύπαρξης του MN στην PCoA.

Το PCoAF είναι πολύ αποδοτικό στην περίπτωση που ο MN απέχει αρκετά από τον HA, καθώς χρησιμοποιεί τον PAR για να προωθεί πακέτα που απευθύνονται στην PCoA. Αυτό εξασφαλίζει έναν ελάχιστο αριθμό χαμένων πακέτων, ειδικά στην περίπτωση κυκλοφορίας multimedia πακέτων, καθώς ο MN σπάνιο επιβεβαιώνει τα ληφθέντα πακέτα.



Εικόνα 16: Previous Care-of-Address Forwarding

Αυτό είναι λειτουργικά παρόμοιο με την τεχνική των Fast Handovers από την άποψη των ελάχιστων χαμένων πακέτων. Ενώ όμως τα fast handovers έχουν καλύτερη απόδοση, μιας και ο MN μπορεί να χρησιμοποιεί την PCoA από τη στιγμή που αντιληφθεί την αλλαγή στον NAR, έχει επιπλέον κόστος για την υποδομή του δικτύου καθώς επίσης και επιπλέον overhead που προκύπτει από την επικοινωνία NAR και PAR. Αντίθετα το PCoAF το μόνο που απαιτεί είναι όλα τα AR να έχουν HA λειτουργικότητα. Εντούτοις με την προσθήκη του ODAD, και των Fast Solicitied Router Advertisements (FSRA) ο μηχανισμός πετυχαίνει αποδόσεις παρόμοιες με αυτές των Fast Handovers.

Ενώ αυτές οι επεκτάσεις των τοπικών AR μπορούν να βελτιώσουν την handover καθυστέρηση, δεν μπορούν ακόμα να λύσουν το πρόβλημα της υπερβολικής σηματοδότησης που εμφανίζεται όταν ο MN κινείται συχνά μέσα σε μια μικρή γεωγραφική περιοχή ενώ βρίσκεται μακριά από το home network του. Με κάθε μετακίνηση, ακόμα και στα όρια κάποιου δικτύου, ο MN πρέπει να στέλνει BU στον

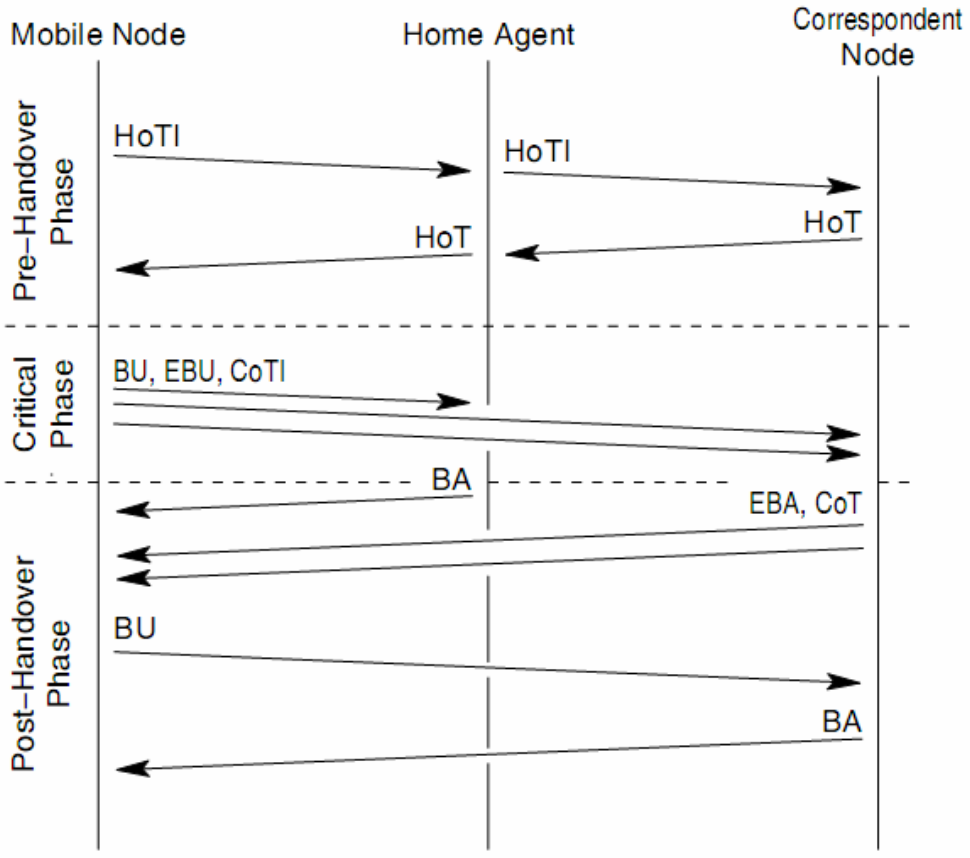
HA του, ακόμα και αν αμέσως μετά επιστρέψει στην προηγούμενη θέση του. Αυτό επιφέρει αυξημένη κίνηση σηματοδοσίας στο δίκτυο. Το MIPv6 δεν σχεδιάστηκε έχοντας αυτό υπόψη δεδομένου ότι ο μόνος σκοπός του ήταν να επιτρέψει σε κινητούς κόμβους πρόσβαση στο Διαδίκτυο. Αυτές οι ανεπάρκειες οδήγησαν στην ανάπτυξη πρωτοκόλλων διαχείρισης τοπικών περιοχών (Localized Mobility Management) που αποτελούν το αντικείμενο του επόμενου κεφαλαίου.

3.7. Early Binding Updates

Η διαδικασία του *return routability* χρησιμοποιείται στο Mobile IPv6 για να ελέγξει την αυθεντικότητα και την εγκυρότητα ενός Binding Update (BU), όπως είδαμε στην παράγραφο 2.3.3.2. Συνοπτικά λοιπόν η *return routability* διαδικασία διενεργεί δύο τεστ. Ένα home-address test (HoT) το οποίο επικυρώνει τον κινητό κόμβο, και ένα care-of-address test (CoT) που ελέγχει την εγκυρότητα της νέας care-of-address.
Η διαδικασία αυτή απαιτεί ελάχιστους πόρους και κόστος μεταξύ των εμπλεκομένων, και δεν εξαρτάται από ακριβές, διαπιστευμένες υποδομές. Εντούτοις, τα δύο τεστ αν και γίνονται παράλληλα, επιφέρουν μια σημαντική καθυστέρηση στην binding διαδικασία, καθώς διενεργούνται ενδεχομένως σε πολύ μεγάλες αποστάσεις.
Η μέθοδος που προτείνεται για την εξάλειψη αυτής της καθυστέρησης είναι τα Early Binding Updates [56]. Σύμφωνα με αυτή τη στρατηγική (Εικόνα 17) τα δύο τεστ δεν πραγματοποιούνται στον κρίσιμο χρόνο που η νέα care-of-address δεν μπορεί ακόμα να χρησιμοποιηθεί. Έτσι ένας κινητός κόμβος εκτελεί το home-address τεστ πριν από κάθε handover. Εάν τα handovers δεν μπορούν να προβλεφτούν, ο κινητός κόμβος μπορεί περιοδικά να επαναλαμβάνει το τεστ. Σε κάθε περίπτωση, ο κινητός κόμβος έχει ένα καινούριο *Home Keygen Token* κάθε φορά που αλλάζει σημείο σύνδεσης και άρα δεν χρειάζεται να κάνει το μακροσκελές home-address τεστ κατά τη διάρκεια της αναπροσαρμογής συνδέσεων. Ο χρόνος ζωής των *Home Keygen Token* ορίζεται ως 3,5 λεπτά και συνεπώς πρέπει να πραγματοποιείται ένα Home-Address Test τουλάχιστον κάθε 3,5 λεπτά. Αντίστοιχα ένα care-of-address τεστ

μπορεί να πραγματοποιηθεί παράλληλα με την αποστολή δεδομένων από και προς την NCoA.

Εισάγονται δύο νέα μηνύματα για την λειτουργία των Early Binding Updates: ένα *Early Binding Update (EBU)* και ένα *Early Binding Acknowledgement (EBack)*. Όταν ο κινητός κόμβος ανιχνεύσει ότι έχει κινηθεί προς ένα διαφορετικό δίκτυο, διαμορφώνει μία νέα CoA και έπειτα στέλνει στον CN ένα Early Binding Update μήνυμα προκειμένου να καταχωρήσει δοκιμαστικά την NCoA με τον CN. Ο κινητός κόμβος χρειάζεται μόνο ένα Home Keygen Token για να επικυρώσει το EBU μήνυμα, ενώ μπορεί να ζητήσει από τον CN να επιστρέψει ένα EBA μήνυμα. Ένας συντηρητικός κινητός κόμβος θα περιμένει το EBA μήνυμα πριν χρησιμοποιεί την νέα CoA του, ενώ ένας αισιόδοξος κινητός κόμβος θα άρχιζε την χρήση της NCoA αμέσως μετά την αποστολή του EBU. Είτε συντηρητικός είτε αισιόδοξος, με τα EBU ο κινητός κόμβος μπορεί να χρησιμοποιήσει την NCoA πιο γρήγορα κατά περίπου ένα round trip time από ό,τι με τα στάνταρ BU.



Εικόνα 17: Early Binding Updates

Στέλνοντας το EBU μήνυμα ο κινητός κόμβος αρχίζει το CoT όπως περιγράφεται στον [17]. Όταν το CoT ολοκληρωθεί, ο MN στέλνει στον CN ένα στάνταρ BU μήνυμα προκειμένου να υποδείξει την αλλαγή από προσωρινή CoA σε μόνιμη CoA.
Από την μεριά του CN, όταν λάβει το EBU μήνυμα από τον κινητό κόμβο, δημιουργεί μια προσωρινή εγγραφή στην binding cache του, και χρησιμοποιεί λοιπόν την NCoA του κινητού κόμβου. Κατά συνέπεια, με τα EBU ο CN μπορεί να χρησιμοποιήσει την NCoA του κινητού κόμβου ένα round trip time πιο γρήγορα από ό,τι με τα στάνταρ biding updates. Η διάρκεια ζωής μιας προσωρινής εγγραφής στην binding cache περιορίζεται σε μερικά δευτερόλεπτα, και προκειμένου να μονιμοποιηθεί απαιτείται η λήψη ενός κανονικού BU. Σε περίπτωση μη λήψης, δηλαδή στην περίπτωση που το home-address test αποτύχει, η εγγραφή σβήνεται.
Η επέκταση των Early Binding Updates είναι πλήρως συμβατή με την κανονική διαδικασία Binding Updates που περιγράφεται στον [17]. Όλα τα μηνύματα σχετικά με τα κανονικά BU παραμένουν αμετάβλητα και διατηρούν την αρχική έννοιά τους. Επιπλέον, ένας κινητός κόμβος μπορεί να αρχίσει την διαδικασία EBU χωρίς γνώση εάν αυτή η βελτιστοποίηση υποστηρίζεται ή όχι από τον αντίστοιχο κόμβο. Εάν ο αντίστοιχος κόμβος δεν την υποστηρίζει το EBU μήνυμα δεν έχει καμία επίδραση, και χρησιμοποιείται η τυποποιημένη διαδικασία.

4. LOCALIZED MOBILITY MANAGEMENT

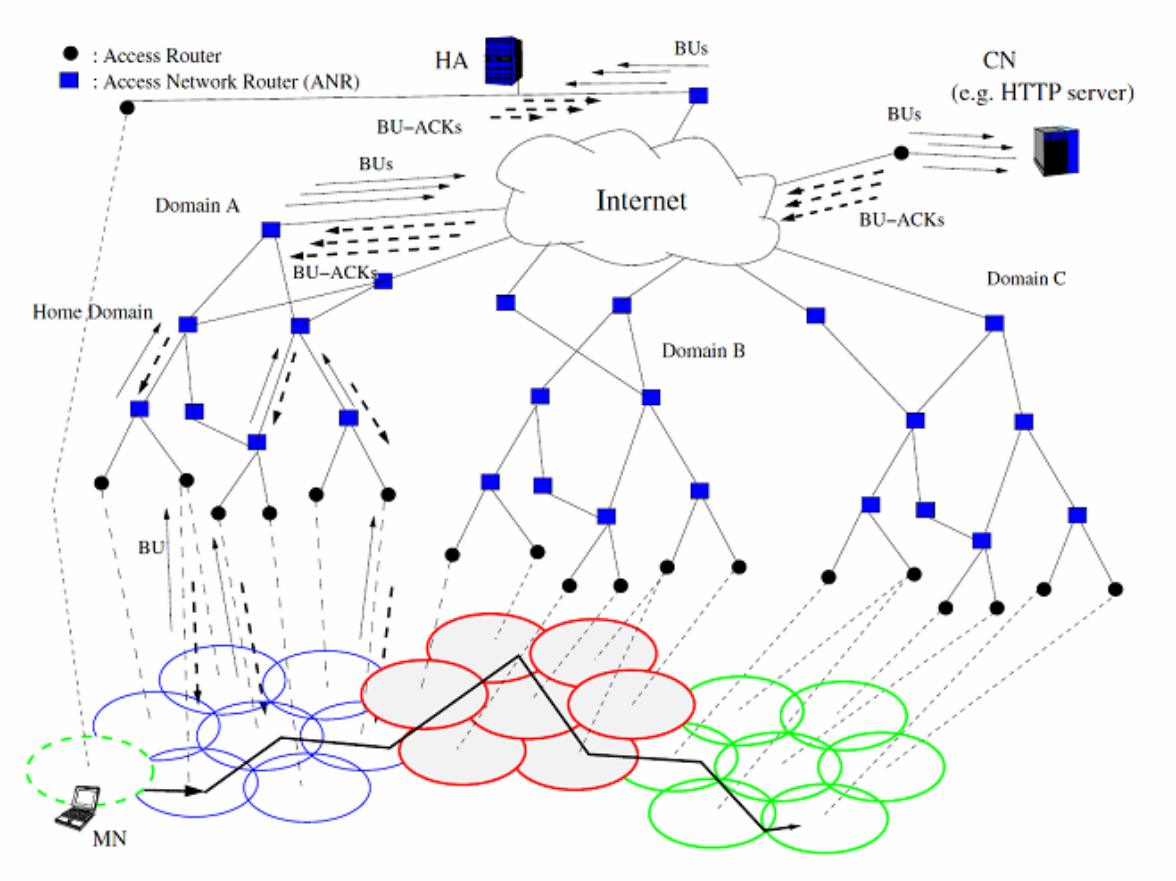
Σαν τοπική κινητικότητα (local mobility) ορίζουμε την κίνηση μέσα σε μία συγκεκριμένη γεωγραφική περιοχή ή μέσα σε μία σαφώς ορισμένη IP-τοπολογία ενός δικτύου. Εδώ και αρκετά χρόνια έχουν ερευνηθεί διάφορα πρωτόκολλα για τη επίλυση του λεγόμενου *Local Mobility Problem* [25], που επικεντρώνονται στη βελτιστοποίηση της διαχείρισης της τοπικής κινητικότητας (localized mobility management). Αυτά τα πρωτόκολλα σχεδιάζονται σε στόχο να μειώσουν την handover καθυστέρηση μετά από σύνδεση σε ένα νέο AR, όπως επίσης και να μειώσουν τον αριθμό των πακέτων σηματοδοσίας μεταξύ του κινητού κόμβου και του ενδεχόμενα απομακρυσμένου Home Agent.

Ας θεωρήσουμε για παράδειγμα έναν κινητό κόμβο που δέχεται real-time πληροφορία (πχ streaming multimedia content) καθώς περιηγείται σε ένα δίκτυο μακριά από το home network του, όπως φαίνεται στην Εικόνα 18. Τα πρωτόκολλα real-time κυκλοφορίας απαιτούν στενά όρια στην παράδοση των πακέτων προκειμένου να κρατηθεί ένα ικανοποιητικό επίπεδο ποιότητας. Καθώς ο MN κινείται μεταξύ των διαφόρων δικτύων σηματοδοτεί τους HA και CN για το νέο σημείο σύνδεσης του με το Internet. Τελικά ο χρόνος ανταλλαγής των BU και των Binding Acknowledgments (BA) θα ξεπεράσει τα όρια αυτά με αποτέλεσμα την απώλεια ποιότητας. Επιπλέον αν θεωρήσουμε το ακόμα χειρότερο σενάριο που ο MN κινείται έντονα προκαλώντας handovers, θα παρουσιαστεί αφενός απώλεια πακέτων λόγο της καθυστερημένης παράδοσης των real-time πακέτων, αφετέρου το δίκτυο θα πλημμυρίσει με BU του κινητού κόμβου προς τους απομακρυσμένους HA και CN.

Για την αντιμετώπιση αυτών των προβλημάτων τα LMM (Localized Mobility Management) πρωτόκολλα εισάγουν την χρήση των Localized Mobility Agent (LMA). Ο LMA τοποθετείται στο επισκεπτόμενο δίκτυο και έχει σαν στόχο να μειώσει την round-trip καθυστέρηση που απαιτείται για την επικοινωνία του MN με τον HA, καθώς και να μειώσει τα πακέτα σηματοδοσίας προς αυτόν.

Ο γενικός μηχανισμός του LMM φαίνεται στην παρακάτω Εικόνα. Ο MN αρχικά ξεκινά από το οικείο δίκτυο του και μεταβαίνει σε ένα νέο δίκτυο. Μια τέτοια κίνηση ονομάζεται *inter-domain*, καθώς ο MN κινείται προς διαφορετικές διαχειριστικές

περιοχές. Ο MN στέλνει BU προς τους HA και CN όπως ακριβώς ορίζει το Mobile IPv6. Ο HA λαμβάνει αυτό το σήμα σαν ένα global mobility σήμα που σηματοδοτεί αυτήν την αλλαγή στο νέο δίκτυο.



Εικόνα 18: Global and Local mobility

Η κίνηση του MN μέσα στο επισκεπτόμενο δίκτυο από υποδίκτυο σε υποδίκτυο ονομάζεται *intra-domain*. Σε αυτό ακριβώς το σημείο εισέρχεται ο LMA. Καθώς λοιπόν η κίνηση του MN σε ένα διαφορετικό υποδίκτυο θα απαιτούσε την αποστολή BU προς τους HA και CN, με την χρήση του LMA αρκεί η αποστολή ενός *regional mobility signal* προς αυτόν. Αυτό το BU είναι γνωστό ως *Regional Binding Update* (RBU). Το LMA είναι έτσι αρμόδιο για τη διατήρηση ενός πίνακα που συνδέει την *Regional Care-of-Address* (RCoA), την οποία γνωρίζουν ο HA και ενδεχομένως κάποιοι CN από τα BU που ο MN στέλνει, με την τρέχουσα θέση του MN, δηλαδή την Local Care-of-Address (LCoA) μέσα στην επισκεπτόμενη περιοχή. Στην πραγματικότητα ο LMA ενεργεί σαν τοπικός HA για αυτή την περιοχή. Συνεπώς το LMM ορίζει ότι η διεύθυνση που ο HA και οι CN χρησιμοποιούν την επικοινωνία με

τον MN να μην αλλάζει όσο ο MN κινείται μεταξύ διαφόρων AR στο ίδιο υποδίκτυο. Με αυτή τη τεχνική λοιπόν, δηλαδή έχοντας τον LMA τοπολογικά κοντά στον MN, το μέγεθος της κυκλοφορίας των BU εξαρτάται αποκλειστικά από το μέγεθος του επισκεπτόμενου δικτύου, το οποίο είναι κατά πολύ μικρότερο από το άγνωστο μέγεθος του Internet.

Τα LMM πρωτόκολλα υπόσχονται να λύσουν τα προβλήματα που περιγράφονται στο [25]. Το [26] δίνει μια πιο λεπτομερής περιγραφή των στόχων που πρέπει να έχουν τέτοιου είδους πρωτόκολλα, καθώς και τρόπους αντιμετώπισης των παρενεργειών που επιφέρουν. Οι στόχοι αυτοί περιγράφονται παρακάτω:

- Βελτίωση απόδοσης handover: Η απώλεια πακέτων κατά το handover εμφανίζεται κατά το χρονικό διάστημα που ο MN δημιουργεί και κατοχυρώνει την IP διεύθυνση του. Κατά τη διάρκεια αυτής της περιόδου, ο κινητός κόμβος δεν είναι προσιτός στην προηγούμενη IP διεύθυνση που οι CN του στέλνουν πακέτα. Σαν στόχος του LMM μπορεί να θεωρηθεί η μείωση του χρόνου που απαιτείται για αλλαγή της διεύθυνσης και την προώθηση της, ώστε να πλησιάζει το άθροισμα της καθυστέρησης που αφορά το L2 handover και την ανίχνευση κίνησης στο επίπεδο 3. Συνεπώς στόχος είναι η μείωση της καθυστέρησης για το L2 και το L3 handover στα 70ms.
- *Μείωση του όγκου πακέτων σηματοδοσίας:* Εάν θεωρήσουμε ότι χρησιμοποιούμε το πρωτόκολλο MIPv6 τότε απαιτούνται κατά μέσο όρο 18 πακέτα σηματοδοσίας για να ολοκληρωθεί επιτυχώς ένα L3 handover σε έναν κόμβο που δεν χρησιμοποιεί route-optimization [26]. Ένας τόσος μεγάλος όγκος πακέτων σηματοδοσίας είναι απαγορευτικός αν αναλογιστούμε καταρχήν ότι ο HA απέχει αρκετά από τον MN, και επιπλέον ότι το ασύρματο φάσμα δεν είναι άπειρο, ούτε δωρεάν. Συνεπώς σκοπός είναι αφενός ο όγκος της σηματοδοσίας να παραμένει στα κατώτερα επίπεδα και αφετέρου το LMM πρωτόκολλο να μην εισάγει επιπλέον.
- *Μυστικότητα θέσης:* Σε οποιοδήποτε IP δίκτυο, υπάρχει η απειλή ένας επιτιθέμενος να καθορίσει τη φυσική θέση ενός κόμβου από την τοπολογική του θέση στο δίκτυο. Η κινητικότητα εισάγει μια πρόσθετη απειλή. Ένας επιτιθέμενος μπορεί να ακολουθήσει τη γεωγραφική θέση ενός κινητού κόμβου, αν ο κινητός κόμβος είναι υποχρεωμένος να αλλάζει IP διεύθυνση

καθώς κινείται από ένα υποδίκτυο προς ένα άλλο, και τελικά να συγκεντρώσει αρκετές πληροφορίες για να βρει τον κόμβο σε πραγματικό χρόνο. Συνεπώς σκοπός είναι να μην απαιτείται η αλλαγή IP διεύθυνσης καθώς ο κόμβος κινείται σε ένα δίκτυο και άρα να μειώνει την πιθανότητα εντοπισμού της θέσης του.

- *Περιορισμός του overhead στο δικτύου:* Τα δίκτυα πρόσβασης έχουν περιορισμούς στο bandwidth. Ειδικά στα ασύρματα δίκτυα ο περιορισμός αυτός έγκειται στον αριθμό των bit/Hz που μπορούν να μεταδοθούν στον αέρα. Επομένως, οποιοδήποτε LMM πρωτόκολλο πρέπει να ελαχιστοποιεί το overhead στο δικτύου πρόσβασης.
- *Απλοποίηση των μεθόδων ασφάλειας των κινητών κόμβων:* Τα LMM πρωτόκολλα απαιτούν επιπλέον μεθόδους ασφαλείας στο συσχετισμό του MN με τον LMA. Αυτή η επιπρόσθετη ασφάλεια απαιτεί επιπλέον σηματοδοσία και συνεπώς επιπλέον χρόνο. Στόχος για τα LMM πρωτόκολλα είναι λοιπόν η επιπλέον απαιτήσεις ασφάλειας να ικανοποιούνται από λειτουργίες που ήδη πραγματοποιούνται σε IP επίπεδο, καταργώντας έτσι την ανάγκη για επιπλεόν σηματοδοσία.
- *Άγνοια link-layer τεχνολογίας:* Ο αριθμός των διαθέσιμων link-layer τεχνολογιών αυξάνει τα τελευταία χρόνια. Συνεπώς ο χρόνος που απαιτείται για την συνεργασία των επιπέδων 2 και 3 είναι μεγάλος. Στόχος λοιπόν των LMM είναι η ανεξαρτητοποίηση από πληροφορίες link-layer για λειτουργίες δρομολόγησης.
- *Υποστήριξη ανομοιόμορφων κινητών κόμβων:* Στην αγορά κυκλοφορούν πολλές συσκευές με διαφορετικά λογισμικά. Ιδανικό θα ήταν σε ένα δίκτυο να μπορούν να εξυπηρετηθούν όσο το δυνατόν περισσότεροι κόμβοι ανεξαρτήτως του λογισμικού που φέρουν. Στόχος λοιπόν είναι η υποστήριξη οποιουδήποτε κόμβου συνδέεται αρκεί να έχει κάποιο interface που να μπορεί να επικοινωνήσει με το δίκτυο.
- *Υποστήριξη για IPv4 και IPv6:* Παρόλο που το IPv6 φαίνεται να κερδίζει έδαφος, είναι επιθυμητό ένα LMM πρωτόκολλο να υποστηρίζει και IPv4.
- *Επαναχρησιμοποίηση υπαρχόντων πρωτοκόλλων:* Υπάρχουν πολλά πρωτόκολλα πάνω στα οποία τα LMM μπορούν να βασιστούν για την

ανάπτυξή τους. Είναι επιθυμητό λοιπόν να χρησιμοποιηθούν ήδη ολοκληρωμένα πρωτόκολλα σε μέρη που η χρήση τους είναι επιτρεπτή.

- *Ανεξαρτητοποίηση Local και Global Mobility Management:* Στόχος είναι η δημιουργία ενός LMM πρωτοκόλλου που δεν θα περιορίζει ή δεν θα περιορίζεται από οποιοδήποτε Global πρωτόκολλο επιλεγεί να χρησιμοποιηθεί.
- *Ρυθμιζόμενη προώθηση μεταξύ Local Mobility Anchor και Mobile Access Gateways:* Διαφορετικοί διαχειριστές δικτύων ίσως απαιτούν διαφορετικούς τύπους προώθησης της κίνησης. Συνεπώς στόχος είναι η επαναρυθμιζόμενη συμπεριφορά των μεθόδων προώθησης.

Ένα άλλο ενδιαφέρον στοιχείο για τα LMM, είναι η ταξινόμηση τους σαν tunneling ή routing [27]. Τα routing πρωτόκολλα παρακολουθούν τους MN και σε κάθε τους handover καταγράφουν στους κατάλληλους δρομολογητές τις αλλαγές αυτές και συνεπώς χρησιμοποιούν συμβατική IP δρομολόγηση. Πίνακες με τέτοιες εγγραφές υπάρχουν σε όλους τους mobility agents στο τοπικό δίκτυο. Παραδείγματα τέτοιων πρωτοκόλλων είναι τα Mobile Multicast Protocol, Handoff-Aware Wireless Access Internet Infrastructure (HAWAII) και Cellular IP. Τα tunneling πρωτόκολλα χρησιμοποιούν τους LMA, οι οποίοι κρατούν πληροφορίες που σχετίζονται με τον AR που ο MN είναι συνδεδεμένος. Έτσι τα πακέτα μεταφέρονται μέσω τούνελ από τον LMA στον MN. Παραδείγματα τέτοιων είναι τα GTP, Hierarchical MIPv6 και BCMP.
Παρακάτω θα περιγράψουμε ένα τέτοιο Tunneling πρωτόκολλο, το Hierarchical MIPv6.

4.1. Αλγόριθμοι Επιλογής Local Mobility Agent

Σε ένα LMM πρωτόκολλο είναι πολύ σημαντικός ο τρόπος με τον οποίο επιλέγεται ο LMA που θα εξυπηρετήσει τον κάθε MN, καθώς επηρεάζει την καθυστέρηση και την συχνότητα των inter-domain handovers. Έχουν προταθεί διάφοροι αλγόριθμοι επιλογής των LMA, που είτε επιτρέπουν στους MN να επιλέγουν αυτόν που τους ταιριάζει καλύτερα ανάλογα με τις ανάγκες τους, είτε αφορούν καθολικά σχήματα σύμφωνα με τα οποία η επιλογή γίνεται από τους ARs.

Ο αλγόριθμος που χρησιμοποιείται στο HMIPv6 που θα μελετήσουμε παρακάτω ονομάζεται *Farthest MAP Selection* [28]. Σύμφωνα με αυτόν, ο MN μαθαίνει την hop απόσταση του κάθε LMA μέσω των RAs που δέχεται και επιλέγει να συνδεθεί με τον απώτατο. Σύμφωνα με αυτή τη λογική, η περιοχή κάλυψης αυτού του LMA είναι μεγαλύτερη από την περιοχή κάλυψης οποιουδήποτε άλλου LMA βρίσκεται από κάτω του και συνεπώς αν υποθέσουμε πως ο MN κινείται με μία σταθερή ταχύτητα, θα προκαλέσει τον λιγότερο αριθμό inter-domain handovers. Από την άλλη μεριά όμως αν όλοι οι MN επιλέγουν τον απώτατο LMA τότε αυτός θα γίνει bottleneck.

Μια άλλη προσέγγιση, η *Velocity-based scheme* [29], εξετάζει την ταχύτητα του κινητού κόμβου για να πάρει αποφάσεις. Η ταχύτητα του κόμβου υπολογίζεται με την βοήθεια των BU που έχει στείλει, διαιρώντας την απόσταση μεταξύ διαδοχικών LMA με την χρονική διαφορά των αντίστοιχων BUs. Στη συνέχεια το δίκτυο ενημερώνει τον MN για την σχετική ταχύτητα όλων των MNs στο συγκεκριμένο LMA έτσι ώστε ο MN να διαλέξει αυτόν που επιθυμεί. Στα πειράματα που έκαναν σύγκριναν τον αλγόριθμο τους με τρεις άλλους, έναν που επιλέγει τον απώτερο, έναν τον κοντινότερο και έναν που επιλέγει τυχαία. Σε όλα τα πειράματα το αλγόριθμος τους είχε την καλύτερη συμπεριφορά.

Ο αλγόριθμος που παρουσιάζεται στο [30] σαν FHMIP-UP χρησιμοποιεί επίσης την ταχύτητα του κινητού κόμβου για την επιλογή του κατάλληλου LMA. Σύμφωνα με αυτόν η ταχύτητα του κόμβου συγκρίνεται με κάποια κατώφλια με στόχο οι πιο γρήγορα κινούμενοι κόμβοι να συνδέονται με LMA που βρίσκονται στα υψηλότερα ιεραρχικά επίπεδα σημειώνοντας έτσι λιγότερα inter-domain handover.

Η ίδια αναφορά κάνει λόγο και για τον αλγόριθμο FHMIP-NH, ο οποίος στηρίζεται στην ιδέα ότι γρήγοροι κόμβοι δεν σημαίνει απαραίτητα ότι κινούνται και εκτός domain. Έτσι αρκεί να μετρηθεί σε πόσο χρόνο ο κάθε κόμβος πραγματοποιεί ένα συγκεκριμένο αριθμό handover[14]. Όσο πιο μικρός αυτός ο χρόνος τόσο υψηλότερου επιπέδου LMA επιλέγεται. Με αυτόν τον τρόπο απελευθερώνονται θέσεις σε LMA για κόμβους που πραγματικά τους χρειάζονται. Μια επέκταση αυτού του αλγορίθμου είναι η χρήση buffer σε κάθε LMA. Ο MN που γνωρίζει ότι πρόκειται να μεταβεί σε νεό LMA ενημερώνει τον παλιό να κρατήσει όποια πακέτα προορίζονται για αυτόν και

[14] Η default τιμή είναι 3 handover

να του τα παραδώσει αργότερα μέσω του καινούριου LMA. Μόνος περιορισμός είναι οι δύο LMA να ανήκουν στο ίδιο domain.
Η μέτρηση όμως της ταχύτητας ενός κινητού κόμβου συχνά δεν είναι καθόλου ακριβείς, και για αυτό προτάθηκε το Mobile Controlled Movement Tracking (MCMT) [31]. Σύμφωνα με αυτό ο κάθε LMA διαφημίζει τις πληροφορίες του σε όλους τους MNs, ο οποίοι με αυτόν τον τρόπο δημιουργεί σιγά σιγά σε δεντρική μορφή την τοπολογία του domain. Έτσι ο MN επιλέγει κάθε φορά τον κοντινότερο LMA, αναζητώντας τον στη λίστα του. Σε συγκριτικά πειράματα έδειξαν πως ο MCMT είναι καλύτερος από τον Farthest MAP, όσο αναφορά την καθυστέρηση και τον φόρτο. Αυτή η μέθοδος όμως απαιτεί αλλαγές στην δομή των RA μηνυμάτων και η απόδοση του εξαρτάται αποκλειστικά από τον χρόνο που χρειάζεται ο MN για να επιλέξει τον κατάλληλο LMA.
Ένας άλλος αλγόριθμος προτάθηκε στο [32] και αναφέρεται σαν *Adaptive Selection Scheme.* Σύμφωνα με αυτό ο MN επιλέγει τον LMA ανάλογα με το Session-to-mobility ratio (SMR). Το SMR είναι ο λόγος του αριθμού νέων LMA που ανακαλύπτονται προς τον αριθμό των handover. Αν η τιμή αυτή ξεπερνά κάποια κατώφλια τότε επιλέγεται ο LMA που ελαχιστοποιεί το κόστος Binding Update και Packet Delivery. Περισσότερες λεπτομέρειες μπορείτε να βρείτε στο [31].

4.2. Hierarchical Mobile IPv6

Το HMIPv6 [28] εισάγει την χρήση ενός νέου είδους κόμβου, του *Mobility Anchor Point* (MAP), που δεν είναι τίποτα άλλο από τον LMA που περιγράψαμε προηγουμένως. Ο MAP παίζει τον ρόλο του HA στο τοπικό δίκτυο και όλες οι κινήσεις του MN μέσα στην δικαιοδοσία του MAP θεωρούνται intra-domain. Ο MAP περιορίζει τον όγκο των πακέτων σηματοδοσίας έξω από το την τοπική περιοχή. Παρουσιάζουμε παρακάτω επιγραμματικά μερικά πλεονεκτήματα του HMIPv6:

- Ο κινητός κόμβος (MN) στέλνει τις BU στον τοπικό MAP και όχι στον HA (που βρίσκεται λογικά πιο μακριά) και στους CNs.
- Μείωση της handover καθυστέρησης κατά ελάχιστο 1,5 round-trip time, επειδή δεν απαιτείται πια η return routability διαδικασία για κάθε CN.

- Μόνο ένα μήνυμα BU απαιτείται να διαβιβαστεί από τον MN προτού η κυκλοφορία επαναδρομολογηθεί από τον HA και όλους CNs στη νέα θέση της. Αυτό είναι ανεξάρτητο από τον αριθμό του CN με τους οποίους ο MN επικοινωνεί.
- Ο MN μπορεί πάντα να χρησιμοποιεί την RCoA ως διεύθυνση προέλευσης κατά την επικοινωνία του με CNs ώστε το route optimization να γίνεται στο global επίπεδο και συνεπώς ο CN να μην μπορεί να παρακολουθεί την μετακίνηση του MN στο MAP domain
- Είναι εύκολο στην υλοποίηση καθώς δεν απαιτούνται αλλαγές στον HA και στους CNs.

Ο MAP είναι απαραίτητα ένας AR, και η λειτουργία του ως MAP αναγνωρίζεται από το MAP πεδίο στα RA. Ένας κινητός κόμβος που μπαίνει σε ένα MAP domain θα λάβει RAs που περιέχουν πληροφορίες για ένα ή περισσότερα τοπικά MAP. Ο MN μπορεί να συνδέσει την τρέχουσα θέση του (LCoA) με μια διεύθυνση στο υποδίκτυο του MAP (RCoA). Ενεργώντας ως HA, ο MAP θα λάβει όλα τα πακέτα εκ μέρους του κινητού κόμβου και θα προωθήσει άμεσα στην τρέχουσα διεύθυνση του κινητού κόμβου. Εάν ο κινητός κόμβος αλλάξει την τρέχουσα διεύθυνσή του (LCoA) μέσα στην τοπική περιοχή του MAP, πρέπει μόνο να την καταχωρήσει στον MAP. Ως εκ τούτου, μόνο η RCoA πρέπει να καταχωρείται στους CNs και στον HA. Η RCoA δεν αλλάζει καθόσον ο MN κινείται μέσα στο MAP domain και συνεπώς η κίνηση του κόμβου είναι αδιαφανής για τους κόμβους που επικοινωνεί.

Τα όρια ενός MAP domain καθορίζονται από τους δρομολογητές (AR), οι οποίοι διαφημίζουν πληροφορίες σχετιζόμενες με τον MAP στους συνδεδεμένους κόμβους. Πρέπει να σημειωθεί ότι το HMIPv6 είναι μια επέκταση στο MIPv6 και συνεπώς επαφίεται στον κάθε κινητό κόμβο αν θα συνδεθεί με τον MAP ή όχι. Εντούτοις, σε μερικές περιπτώσεις ο κινητός κόμβος μπορεί να προτιμήσει να χρησιμοποιήσει το απλό MIPv6 πρωτόκολλο, πχ στην περίπτωση που βρίσκεται σε ένα υποδίκτυο του δικού του δικτύου, και άρα να συνδεθεί με κάποιον HA που βρίσκεται κοντά του.

Παρόλα τα πολυάριθμα πλεονεκτήματα όμως που περιγράψαμε παραπάνω το HMIPv6 έχει επίσης μειονεκτήματα. Τα σημαντικότερα περιγράφονται παρακάτω:

- η μη-βέλτιστη δρομολόγηση που παρέχει το HMIPv6 δεν συμφωνεί με την απαίτηση εξελιξιμότητας στα LMM πρωτόκολλα.

- ο μηχανισμός ανακάλυψης MAP είναι εξ ορισμού μια χειρωνακτική διαδικασία επειδή τα MAPs και τα ARs πρέπει να διαμορφωθούν πριν τη χρήση τους. Η διαμόρφωση περιλαμβάνει τον ορισμό των AR που θα λειτουργούν ως MAP και τον ορισμό των AR που θα διαφημίζουν το συγκεκριμένο MAP και συνεπώς θα ορίζουν και τα όρια του. Για μεγάλα δίκτυα αυτό απαιτεί πολύ σημαντικό φόρτο εργασίας.
- Υπό συγκεκριμένες περιπτώσεις το HMIPv6 μπορεί να επιβάλει σημαντικό overhead, χρησιμοποιώντας μέχρι και δύο σετ τούνελ επικεφαλίδων σε κάθε κατεύθυνση. Εφόσον ο MN δεν μπορεί να εκτελέσει σύμφωνα με το [28] βελτιστοποίηση διαδρομών με τον CN, τότε εγκαθιδρύονται τούνελ μεταξύ της RCoA και του HA. Το ίδιο αποτέλεσμα έχουμε και στην περίπτωση που ο MN δεν επιθυμεί ο CN να γνωρίζει την RCoA του.
- Δεδομένου ότι ο MAP πρέπει να κρατήσει το αρχικό πακέτο άθικτο θα υπάρχει πάντα ένα τούνελ από το MAP στο MN, με σημείο εισόδου την διεύθυνση του MAP και σημείο εξόδου την LCoA του MN. Ενώ είναι δυνατό να καταργήσουμε αυτό το τούνελ εφαρμόζοντας βελτιστοποίηση διαδρομής από τον CN στον MN τότε ο CN θα γνωρίζει τη θέση του MN, κάτι το οποίο αντιτίθεται στο σκοπό του HMIPv6.
- Ο αλγόριθμος επιλογής MAP δεν είναι βέλτιστος καθώς ορίζει ότι θα επιλεγεί ο απώτερος στην ιεραρχία, και συνεπώς ο ίδιος MAP θα επιλεγεί από όλους τους MN που βρίσκονται κάτω από αυτόν. Κατά συνέπεια ο MAP θα είναι bottleneck. Στο [28] ορίζεται η δυνατότητα κάθε MAP να έχει μία τιμή προτίμησης, έτσι ώστε αν ο MAP είναι συνωστισμένος να μπορεί να την μειώνει και να προτιμούνται γειτονικοί MAPs. Αυτό όμως απλά θα μεταφέρει το πρόβλημα σε κάποιον άλλο MAP. Πρέπει λοιπόν να υπάρχει ένα δυναμικό σχήμα εξισορρόπησης φορτίου σε ολόκληρο το domain που να που να μεταβάλει και τα όρια των MAP εκτός από την προτίμηση τους.

4.3. Fast Handovers for Hierarchical MIPv6

Μια πολύ φυσική και απλή επιλογή θα ήταν να συνδυάσουμε το FMIPv6 με το HMIPv6, δηλαδή αρκεί να εφαρμόσουμε την FMIPv6 στα HMIPv6 δίκτυα [33]. Κάτι τέτοιο δεν θα ήταν αποδοτικό καθώς εμφανίζει τα ακόλουθα μειονεκτήματα:

- Ο PAR πρέπει να διεκπεραιώνει εκτός από το tunneling των πακέτων από τον MAP στον MN όπως ορίζει το HMIPv6, και τις handover διαδικασίες για τα fast handover όπως ορίζει το FMIPv6. Για να γίνει αυτό ο PAR πρέπει πρώτα να παίρνει τα πακέτα που πηγαίνουν προς τον MAP, προσθέτοντας έτσι επιπλέον overhead στο HMIPv6.
- Στο HMIPv6, το πραγματικό μονοπάτι του αμφίδρομου τούνελ μεταξύ PAR και NAR μπορεί να περιλαμβάνει και τον MAP (δηλαδή, PAR-MAP-NAR). Συνεπως τα πακέτα θα διασχίζουν την διαδρομή μεταξύ των ARs και του MAP δύο φορές. Αυτό επιφέρει άσκοπη κατανάλωση bandwidth.
- Κατά τη διάρκεια του handover, η πιθανότητα τα tunneled πακέτα από τον PAR στον NAR πριν το FBU να φτάσουν αργότερα από στον NAR από ατά που στέλνονται κατευθείαν στον MAP μετά το FBU είναι αρκετά υψηλή. Κάτι τέτοιο θα προκαλέσει λανθασμένη σειρά άφιξης των αριθμημένων πακέτψν και συνεπώς χαμηλή απόδοση.

Συνδυάζοντας όλα αυτά, η συνολική handover καθυστέρηση και το tunneling overhead θα παρουσιαστούν ιδιαίτερα αυξημένα. Συνεπώς μια τέτοια προσέγγιση δεν μπορεί να αξιοποιήσει ταυτόχρονα τα πλεονεκτήματα των FMIPv6 και HMIPv6.

Η λύση που δίδεται λοιπόν για την τεχνική Fast Hierarchical MIPv6 (F-HMIPv6), είναι το τούνελ να σχηματίζεται μεταξύ MAP και NAR, και όχι μεταξύ PAR και NAR [34, 35]. Για αυτόν το λόγο, ο MN ανταλλάσσει τα μηνύματα σηματοδοσίας με τον MAP και όχι με τον PAR. Το F-MIPv6 χρησιμοποιεί τα μηνύματα που χρησιμοποιεί το FMIPv6 για την handover υποστήριξη, χωρίς να εισάγει κανένα επιπλέον τύπο μηνύματος.

Η διαδικασία του F-HMIPv6 παρουσιάζεται στην Εικόνα 19. Υποθέτουμε πως ένας κινητός κόμβος προσπαθεί να κινηθεί από τον PAR προς τον NAR μέσα στο MAP domain, και ότι ο MAP έχει ήδη πληροφορίες για την link-layer διεύθυνση και το πρόθεμα κάθε AR στην περιοχή. Επίσης, υποτίθεται ότι το επίπεδο 2 ενημερώνει για

ενδεχόμενα handover τον κινητό κόμβο. Παρατηρούμε επίσης πως όπως φαίνεται στην εικόνα, το F-HMIPv6 χρησιμοποιεί ακριβώς τον ίδιο αριθμό μηνυμάτων με το FMIPv6.

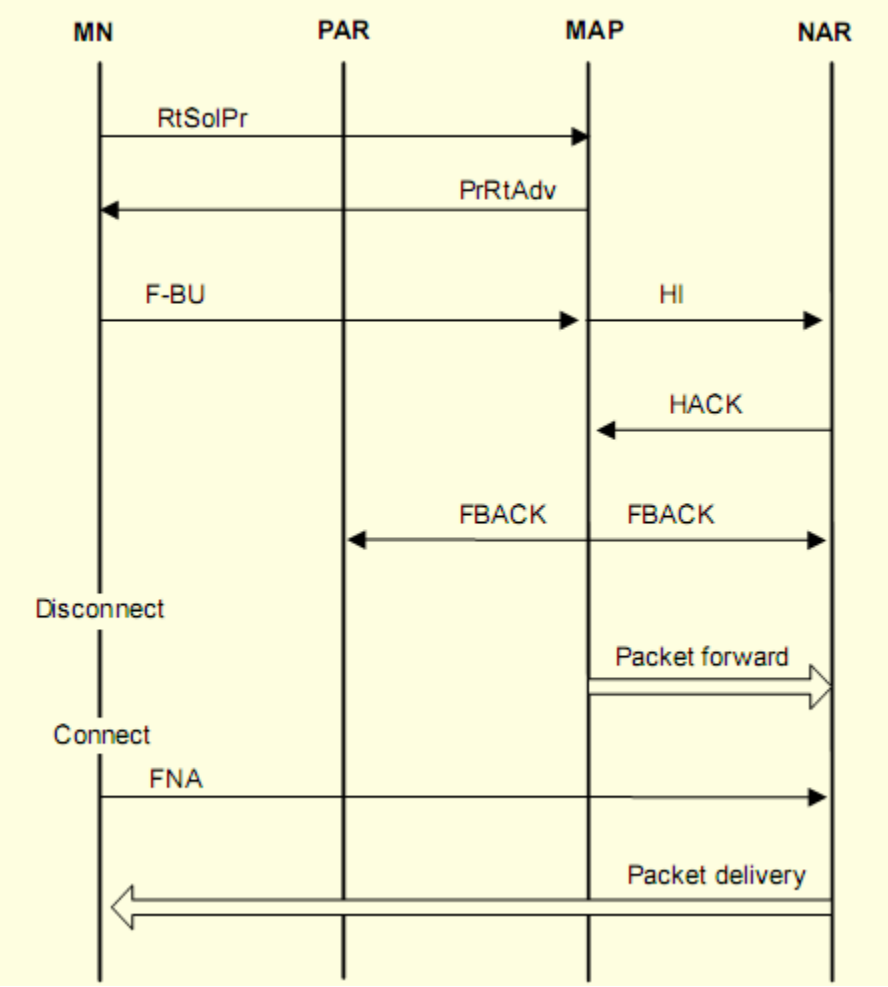
Όταν λοιπόν ο MN ξέρει πως θα κινηθεί προς τον NAR, λόγω του L2 trigger, στέλνει ένα Router Solicitation for Proxy (RtSolPr) για να ζητήσει πληροφορίες για τον NAR και μια νέα διεύθυνση (LCoA). Μόλις ο MAP λάβει αυτό το μήνυμα, θα απαντήσει με Proxy Router Advertisement (PrRtAdv) που θα περιέχει το πρόθεμα του NAR.

Ο MN διαμορφώνει κατόπιν την νέα του CoA χρησιμοποιώντας της πληροφορίες προθέματος που έλαβε. Έπειτα ο MN στέλνει ένα Fast Binding Update (FBU) στον MAP που πειρέχει την νέα του LCoA. Με τη σειρά του ο MAP ξεκινά τις διαδικασίες fast handover στέλνοντας το μήνυμα Handover Initiate (HI) στον NAR. Με αυτό το μήνυμα επαληθεύεται η νέα LCoA και καθιερώνεται ένα αμφίδρομο τούνελ μεταξύ MAP και NAR. Ο NAR εκτελεί DAD και απαντά στον MAP με Handover Acknowledgement (HACK). Μόλις ο MAP λάβει το HACK απαντά στον MN με ένα Fast Binding Acknowledgement (FBack).

Όταν τελικά ο MN συνδεθεί με τον NAR, στέλνει μήνυμα Router Solicitation (RS) που περιέχει και ένα μήνυμα Fast Neighbor Advertisement (FNA) προκειμένου να ενημερώσει για την παρουσία του. Από αυτό το σημείο και μετά ο NAR προωθεί όλα τα πακέτα στον MN.

Όπως αναφέραμε και νωρίτερα, η F-HMIPv6 τεχνική δεν χρησιμοποιεί νέα μηνύματα εκτός από αυτά που έχουν ήδη καθοριστεί στα HMIPv6 και FMIPv6. Εντούτοις, απαιτούνται οι ακόλουθες επεκτάσεις των υπαρχόντων μηνυμάτων:

- Ένα νέο flag ορίζεται για τον MAP που να υποδεικνύει αν ο συγκεκριμένος MAP υποστηρίζει ή όχι F-HMIPv6 μέσα στην περιοχή του.
- Τα FMIPv6 μηνύματα RtSolPr, PrRtAdv, FBU, HI, HACK και FBack έχουν διαφορετικές IP διευθύνσεις στα πεδία διευθύνσεων προέλευσης ή προορισμού. Ειδικότερα, χρησιμοποιείται η διεύθυνση MAP αντί της διεύθυνσης PAR.



Εικόνα 19: Η διαδικασία του F-HMIPv6

4.4. Άλλα LMM πρωτόκολλα

4.4.1. Cellular IP

Στο Cellular IP (CIP) [36] η κάθε περιοχή κάλυψης ονομάζεται κυψέλη, ενώ τα AP Base Stations (BS). Τα BS ανά τακτά χρονικά διαστήματα μεταδίδουν RA, τα οποία χρησιμοποιούν οι κινητοί χρήστες για να προσδιορίσουν τον κοντινότερο BS. Κάθε πακέτο που μεταδίδουν οι MN δρομολογείται μέσω του BS, ενώ τα ίδια τα πακέτα

χρησιμοποιούνται από όλους τους κόμβους προκειμένου να εμπλουτίζουν την Route cache τους. Με αυτόν τον τρόπο μειώνονται τα πακέτα σηματοδοσίας, ενώ παράλληλα οι κόμβοι ανακαλύπτουν μονοπάτια για την δρομολόγηση πακέτων. Το Cellular IP έχει δύο τεχνικές για την αντιμετώπιση των handover. Η hard handover τεχνική θυσιάζει κάποια πακέτα εις όφελος της μειωμένης σηματοδοσίας και της απλότητας. Με την semisoft handover τεχνική επιτρέπεται ο MN να λαμβάνει πακέτα και από τον PAR και από τον NAR ταυτόχρονα. Με αυτό τον τρόπο ελαχιστοποιείται η απώλεια πακέτων και συνεπώς παρέχει καλύτερη απόδοση από την hard τεχνική με αυξημένη όμως πολυπλοκότητα. Υπάρχει επίσης διάκριση μεταξύ ενεργών και μη-ενεργών χρηστών με στόχο την εξοικονόμηση ενέργειας. Όταν υπάρχουν πακέτα με προορισμό έναν μη-ενεργό χρήστη τότε αυτός πρώτα καλείται με ένα broadcast μήνυμα και γίνεται ενεργός. Το Cellular IP επίσης προσφέρει μηχανισμούς ασφαλείας κατά το handover χωρίς να εισάγει επιπλέον σηματοδοσία, αλλά χρησιμοποιώντας ήδη υπάρχοντα κλειδιά από τους BS.

4.4.2. Handoff-Aware Wireless Access Internet Infrastructure

Το πρωτόκολλο Handoff-Aware Wireless Access Internet Infrastructure (HAWAII) [37] βασίζεται στο Mobile IP για την παροχή inter-domain κινητικότητας. Ένας κινητός κόμβος που μπαίνει σε ένα νέο υποδίκτυο του ανατίθεται μια collocated CoA (coCoA), ενώ παράλληλα διατηρεί την παλιά CoA. Συνεπώς όσο ο MN κινείται σε αυτό το domain δεν χρειάζεται να αλλάξει την διεύθυνση του και άρα ο HA δεν γνωρίζει καν τις μετακινήσεις του. Οι κόμβοι σε ένα HAWAII δίκτυο διατηρούν πληροφορίες δρομολόγησης για κάθε άλλο κόμβο σε routing tables. Αυτές οι πληροφορίες λαμβάνονται μέσω μηνυμάτων σηματοδοσίας που περιοδικά στέλνουν οι κινητοί κόμβοι. Το HAWAII καθορίζει τέσσερα εναλλακτικά σχήματα για την αντιμετώπιση των handover. Έτσι ανάλογα με τις προτεραιότητες των χρηστών, ελαχιστοποίηση απώλειας πακέτων, ελαχιστοποίηση handover καθυστέρησης και διατήρηση αρίθμησης πακέτων, επιλέγεται αυτό που ταιριάζει περισσότερο. Το

HAWAII χρησιμοποιεί IP multicasting προκειμένου να παραδώσει πακέτα σε κόμβους για τους οποίους δεν υπάρχει καμία πληροφορία στις routing caches.

4.4.3. Intra-domain Mobility Management Protocol

Το Intra-domain Mobility Management Protocol (IDMP) [38] είναι ένα πρωτόκολλο κινητικότητας που υποστηρίζει τη δρομολόγηση πακέτων προς κινητούς κόμβους μέσα σε μία περιοχή κινητικότητας. Μια περιοχή κινητικότητας ορίζεται σαν μια συλλογή IP υποδικτύων που αθροίζονται με βάση κάποιους παράγοντες όπως η γεωγραφική εγγύτητα. Το IDMP υποστηρίζει fast handovers με ελάχιστες απώλειες πακέτων και paging για μειωμένη σηματοδοσία. Επίσης χρησιμοποιεί ιεραρχική δομή, με έναν LMA στην κορυφή και διάφορους FA στα κλαδιά του. Το κορυφαίο επίπεδο λειτουργεί σαν gateway στο Διαδίκτυο, ενώ δεν χρειάζεται καμία καθολική ενημέρωση όσο ο MN κινείται στα όρια του domain. Διατηρούνται δύο διευθύνσεις, η καθολική που είναι συνδεδεμένη με τον κορυφαίο LMA και αλλάζει μόνο στην περίπτωση που ο MN βγει εκτός domain, και η τοπική η οποία αλλάζει κάθε φορά ο κινητός κόμβος κάνει handover σε κάποιον γειτονικό FA εντός του domain.

4.4.4. Edge Mobility Architecture

Το Edge Mobility Architecture (EMA) [40] ορίζει ένα γενικό πλαίσιο για τη διαχείριση κινητικότητας μέσα σε ένα ασύρματο domain. Μέσα σε αυτό το πλαίσιο, είναι δυνατό, θεωρητικά, να χρησιμοποιηθεί οποιοδήποτε πρωτόκολλο δρομολόγησης για να διαβιβάσει τα πακέτα. Οι συντάκτες προτείνουν το Temporally Ordered Routing Algorithm (TORA) [41] Αυτή η επιλογή φαίνεται να εξασφαλίζει μια καλή εξελιξιμότητα για το σύστημα ενώ η EMA αρχιτεκτονική επιτρέπει την υιοθετήσει του TORA για την διαχείριση ασύρματων δικτύων που έχουν άλλες ιδιότητες από τα ad-hoc. Το EMA ορίζει το handover σαν μια απολύτως διαφανή διαδικασία για τα ανώτερα στρώματα, ακόμη και στο πρωτόκολλο δρομολόγησης. Αυτός ο μηχανισμός βασίζεται στην λήψη ενός *Strong Handoff Radio Trigger* (SHRT) μηνύματος από τον MN για να αρχίσει την handover διαδικασία. Με αυτό δημιουργούνται τούνελ μεταξύ

του MN και των PAR, NAR, τα οποία αναλαμβάνουν την μεταφορά των πακέτων προς τον MN έως ότου βρεθούν εναλλακτικά μονοπάτια προς αυτόν. Αυτό επιτυγχάνεται με τη χρησιμοποίηση των κανονικών μηχανισμών του επιλεγμένου πρωτοκόλλου δρομολόγησης. Οι προσωρινές σήραγγες αφαιρούνται με χρήση δύο τεχνικών: Break before Make και Make before Break που έχουν περιγράφει παραπάνω. Όσο αναφορά τη δρομολόγηση το EMA υποστηρίζει δύο τύπους: την βασισμένη στο πρόθεμα δρομολόγηση (όπως στα κλασσικά IP δίκτυα) και δρομολόγηση ανάλογα με τον κόμβο. Όσο ο MN είναι μέσα στο domain η δρομολόγηση γίνεται κανονικά μέσω του προθέματος. Όταν ο MN αλλάξει domain χρησιμοποιούνται συγκεκριμένα δρομολόγια προς τους CNs.

5. Προσομοιωση Mobile IPv6 Επεκτασεων

5.1. Ο προσομοιωτής OMNeT++

Ο προσομοιωτής OMNeT++ [42] είναι ένα open-source, αντικειμενοστραφές, αρθρωτό, ανοιχτής αρχιτεκτονικής περιβάλλον προσομοίωσης με πολύ ισχυρό γραφικό περιβάλλον. Ο βασικός τομέας εφαρμογής του είναι η προσομοίωση των επικοινωνιακών δικτύων, αλλά λόγο της γενικής και εύκαμπτης αρχιτεκτονικής του, έχει επιτυχώς χρησιμοποιηθεί και σε άλλες περιοχές, όπως στην προσομοίωση IT[15] συστημάτων, δικτύων αναμονής, hardware αρχιτεκτονικών και επιχειρησιακών διαδικασιών. Βασικό πλεονέκτημα είναι επίσης η πολύ καλή τεκμηρίωση του.

Ένα μοντέλο στο OMNeT++ απεικονίζεται από ιεραρχικά εμφωλευμένα στοιχεία. Το βάθος του κάθε στοιχείου δεν είναι περιορισμένο, και έτσι δίνεται η δυνατότητα στο χρήστη να απεικονίσει τη λογική δομή οποιουδήποτε πραγματικού συστήματος στην πρότυπη δομή του. Τα στοιχεία αυτά επικοινωνούν μεταξύ τους μέσω μηνυμάτων., τα οποία μπορούν να περιέχουν αυθαίρετα σύνθετες δομές δεδομένων. Τα μηνύματα μπορούν να σταλούν απευθείας στον προορισμό τους, είτε κατά μήκος μιας προκαθορισμένης πορείας, μέσω πυλών και συνδέσεων. Το κάθε στοιχείο μπορεί να έχει τις δικές του παραμέτρους οι οποίες χρησιμοποιούνται για την εξατομίκευση της συμπεριφοράς του στοιχείο στην τοπολογία.

Το OMNeT++ γίνεται όλο και πιο δημοφιλές στην επιστημονική κοινότητα και στην βιομηχανία. Επίσης γίνεται μια συνεχής προσπάθεια εμπλουτισμού του με την προσθήκη διαφόρων επεκτάσεων (MPLS, IPv6, WDM κτλ) προκειμένου να καλυφτεί όλο το επιστημονικό φάσμα.

[15] IT: Information Technology

5.2. IPv6Suite Simulation Framework

Το IPv6Suite [44] είναι μια open-source επέκταση για τον προσομοιωτή OMNeT++ η οποία επιτρέπει την ακριβή προσομοίωση IPv6 πρωτοκόλλων και δικτύων. Επεκτείνει το INETFramework [45] προσθέτοντας λειτουργικότητα των ακόλουθων RFCs:

- RFC 2373 IP Version 6 Addressing Architecture [14]
- RFC 2374 An IPv6 Aggregatable Global Unicast Address Format [46]
- RFC 2460 Internet Protocol, Version 6 (IPv6) Specification [11]
- RFC 2461 Neighbor Discovery for IP Version 6 (IPv6) [13]
- RFC 2462 IPv6 Stateless Address Autoconfiguration [16]
- RFC 2463 Internet Control Message Protocol (ICMPv6) for the Internet Protocol Version 6 (IPv6) Specification [47]
- RFC 2464 Transmission of IPv6 Packets over Ethernet Networks [48]
- RFC 2472 IP Version 6 over PPP [49]
- RFC 2473 Generic Packet Tunneling in IPv6 [12]
- RFC 3775 Mobility Support in IPv6 (no security) [17]

και των ακόλουθων Internet Drafts:

- Hierarchical Mobile IPv6 Mobility Management (HMIPv6), revision 2 [50]
- Optimistic Duplicate Address Detection, revision 0 [51]
- Fast Solicited Router Advertisements, revision 4 [52]
- Early Binding Updates for Mobile IPv6, revision 0 [53]
- Neighbor Discovery for IP version 6 - for Fast Router Solicitations [54]

Εισάγει επίσης στοιχεία για την προσομοίωση ασύρματων LAN, που ικανοποιούν τα ακόλουθα standards:

- IEEE 802.3
- IEEE 802.11b

Ο στόχος του IPv6Suite [55] συγκεντρώνεται στο να παρέχει ένα όσο το δυνατόν πιο ακριβές εργαλείο, χωρίς να θυσιάσει την απόδοση ή την εξελιξιμότητα. Έπρεπε να είναι εξελικτικό επειδή προγραμματίζεται να χρησιμοποιηθεί ως πλατφόρμα προσομοίωσης ρεαλιστικών δικτύων, πολύ μεγάλης κλίμακας, προκειμένου να γίνει κατανοητό πόσο καλά λειτουργούν οι νέες τεχνολογίες, όπως το MIPv6, όταν δοκιμάζονται σε δίκτυα πραγματικών διαστάσεων.

5.3. Το μοντέλο εξομοίωσης

Για την μελέτη μας πάνω στην handover καθυστέρηση μελετήθηκαν διάφορα σενάρια, με κύριο στόχο την επιλογή ενός μοντέλου που αφενός να είναι ρεαλιστικό και αφετέρου αρκούντως περίπλοκο! Η Εικόνα 20 δείχνει την τοπολογία του δικτύου, όπως αυτή παρουσιάζεται από το OMNet++. Ο κινητός κόμβος ξεκινά από το home agent του και κινείται προς το foreign network με διάφορες ταχύτητες διασχίζοντας συνολικά 10 διαφορετικά υποδίκτυα, ένα ανά 150 μέτρα.

Το δίκτυο φαίνεται στην Εικόνα 20 μοιάζει να είναι αρκετά απλουστευμένο, κάτι όμως που δεν είναι απόλυτα αληθές. Αφενός δεν θέλαμε μία πολύ μεγάλη και περίπλοκη τοπολογία, καθώς δεν είναι ούτε ο σκοπός του πειράματος μας, ούτε θα ευνοούσε τις μετρήσεις μας, και άρα πρέπει το μοντέλο μας να είναι όσο το δυνατόν πιο αφαιρετικό. Αφετέρου η συγκεκριμένη τοπολογία μπορεί να εξομοιώσει επιτυχώς μια ρεαλιστική περίπτωση που ο κινητός κόμβος διασχίζει το Διαδίκτυο επικοινωνώντας με κάποιον απομακρυσμένο κόμβο. Ας δούμε καλύτερα πως μπορεί να γίνει αυτό. Οι συνδέσεις και οι αντίστοιχές καθυστερήσεις που επιβάλουν μπορούν να θεωρηθούν ως αφηρημένη και απλουστευμένη έκδοση της Διαδυκτιακής καθυστέρησης ακόμα κι αν στην προσομοίωση σας παρουσιάζονται ως μία απευθείας σύνδεση. Έτσι αντιμετωπίζουμε τις πραγματικές καθυστερήσεις συνδέσεων σαν καθυστερήσεις διάδοσης στην προσομοίωση, και συνεπώς τις θεωρούμε ως καθυστέρηση διάδοσης του κάθε πακέτου που ταξιδεύει στο μονοπάτι. Αυτή η απλή θεώρηση μπορεί να δώσει αποτελέσματα που είναι επαρκή στη σύγκριση των σχετικών οφελών κάθε τεχνολογίας που εξετάζεται.

Επιπλέον πρέπει να τονίσουμε ένα πολύ μεγάλο πλεονέκτημα του σεναρίου που τελικά επιλέχτηκε. Είναι δυνατόν να εφαρμοστεί σε κάθε μία από τις εξεταζόμενες τεχνολογίες με ελάχιστες ή καθόλου αλλαγές. Πιο συγκεκριμένα η μόνη αλλαγή που απαιτείται είναι η χρήση του router ως MAP (βλέπε Εικόνα 20) για την περίπτωση που εξετάζουμε την HMIPv6 τεχνολογία. Αυτό είναι ένα πολύ σημαντικό όφελος της συγκεκριμένης τοπολογίας καθώς όλες οι τεχνολογίες εξετάζονται στις ίδιες συνθήκες και προδιαγραφές κάνοντας έτσι πολύ εύκολη και λογική την σύγκριση τους.

Παρακάτω παρουσιάζονται τα τεχνικά χαρακτηριστικά του μοντέλου μας.

Αντικείμενο	**Παράμετρος**	**Τιμή**
IEEE 802.11	Bandwidth	100Mbps
	Beacon Period	0.1 sec
	Authentication Wait Entry Timeout	2 sec
	Authentication Entry Timeout	2 sec
	Association Entry Timeout	120 sec
	Failed Retransmit Limit	3
	Tx Power	1.5 Watt
	Receiving Threshold Power	-96 dBm
	Handover Threshold Power	-90 dBm
	Authentication Timeout	2 sec
	Association Timeout	2 sec
	Retry	7
IPv6	Link MTU	1280 octets
	Duplicate Address Detection Transmits	1
	Retransmit Timer	1 sec
MIPv6	MIPv6MaxRtrAdvInterval*	1.5 sec
	MIPv6MinRtrAdvInterval*	1 sec
	MaxConsecMissedRtrAdv[16]	1

* εκτός από την περίπτωση των Fast RA beacons

Σε κάθε εκτέλεση των πειραμάτων μας μετράμε τον αριθμό των χαμένων πακέτων και τη handover καθυστέρηση.

Σε κάθε πείραμα ο κινητός κόμβος *client* κινείται από την αρχική του θέση κατά μήκος των 10 access points, ενώ ταυτόχρονα στέλνει αριθμημένα Internet Control Message Protocol (ICMP) ping μηνύματα προς τον απομακρυσμένο κόμβο *server* κάθε 10 msec. Ο *server* δεν επιστρέφει acknowledgment μηνύματα στον *client,* απλά καταγράφει την άφιξη των μηνυμάτων. Με τη βοήθεια αυτών των στατιστικών μπορούμε να μετρήσουμε την απώλεια των πακέτων και κατά συνέπεια και τον

[16] Μέγιστος αριθμός χαμένων Router Advertisements πριν από L2 trigger

χρόνο κατά τον οποίο ο κινητός κόμβος δεν μπορούσε να στείλει δεδομένα. Πιο συγκεκριμένα σαν χαμένα πακέτα εννοούμε τα πακέτα p_2-p_1, όπου p_1 είναι ο αριθμός ακολουθίας του τελευταίου πακέτου που έφτασε στον CN πριν το handover και p_2 είναι ο αριθμός ακολουθίας του πρώτου πακέτου που έφτασε στον CN μετά το handover. Πακέτα που φτάνουν με λανθασμένη αρίθμηση απλά αγνοούνται.
Σαν handover καθυστέρηση θεωρούμε το ελάχιστο μεταξύ του χρόνου μεταξύ των χαμένων πακέτων που μετρήσαμε παραπάνω, και τον χρόνο που παρεμβλήθηκε μεταξύ του L2 handover και της λήψης ενός νέου Binding Acknowledgment (BA) μηνύματος που επιβεβαιώνει την νέα care-of-address (Εικόνα 13). Εξαίρεση αποτελούν η περίπτωση του HMIPv6 που σταματάμε να μετράμε όταν ληφθεί το LBA (Local Binding Acknowledgment) και η περίπτωση του ODAD που μετράμε έως την λήψη του BU καθώς ο κόμβος αρχίζει να χρησιμοποιεί την NCoA χωρίς επιβεβαίωση .
Κατά τις μετρήσεις μας δεν λαμβάνουμε υπόψη μας το πρώτο handover. Μελετάμε λοιπόν μόνο τα handover που γίνονται κάτω από τον ar (Εικόνα 20). Με αυτό τον τρόπο απομονώνουμε διάφορους απρόβλεπτους παράγοντες και τυχόν αρχικοποιήσεις, μελετώντας έτσι ακριβέστερα τα πλεονεκτήματα της κάθε τεχνικής. Έτσι λοιπόν σε κάθε τρέξιμο του πειράματος καταγράφουμε και μελετάμε 9 handover.

5.4. Τεχνικές υπό εξέταση

Στα πειράματα μας θα μελετήσουμε τις περισσότερες από τις τεχνικές που περιγράφηκαν στα κεφάλαια 3 και 4. Πιο συγκεκριμένα θα εξετάσουμε τις παρακάτω τεχνικές τις οποίες περιγράφουμε ξανά πολύ συνοπτικά:

- Mobile IPv6

  Κανονικές λειτουργίες του MIPv6 όπως έχουν προταθεί από το RFC 3775
- Fast RA beacons

  Αποστολή συχνών RA, μεταξύ 30ms και 70ms
- Fast Solicited Router Advertisements

  Άμεση απάντηση router σε RS

- Early Binding Updates

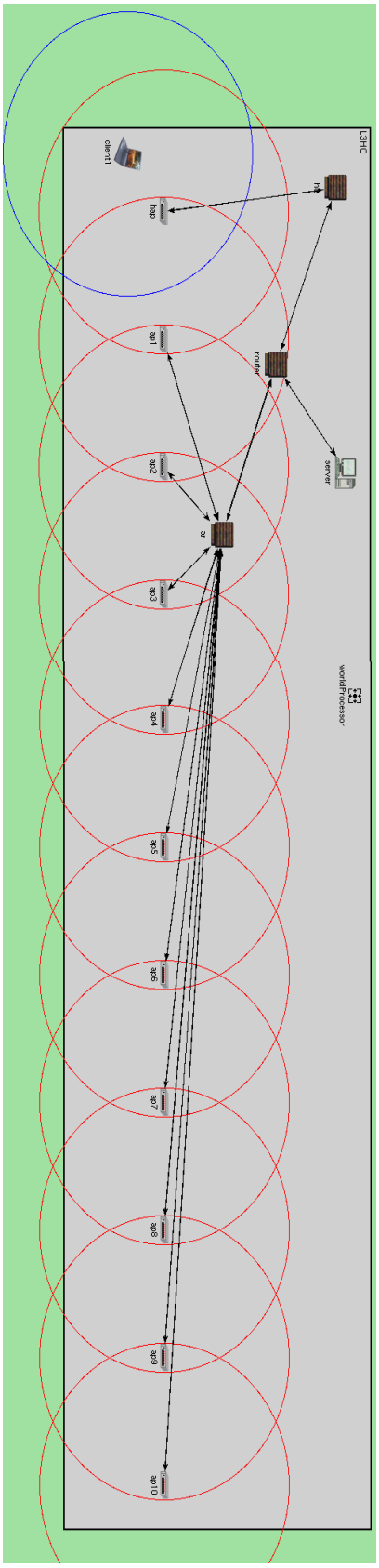
  Αποστολή EBU μηνυμάτων για ταχύτερο binding
- L2 triggers

  Ενημέρωση για ενδεχόμενο L3 handover από το 2^{ο} επίπεδο
- Optimistic Duplicate Address Detection

  Χρήση Optimistic DAD τεχνικών
- Hierarchical Mobile IPv6

  Χρήση της ιεραρχικής δομής του μοντέλου όπως ορίζεται στο RFC 4140

Θα μελετήσουμε επίσης και κάθε δυνατό και επιτρεπτό συνδυασμό μεταξύ τους, 46 περιπτώσεις στο σύνολο τους!



Εικόνα 20: Το μοντέλο εξομοίωσης

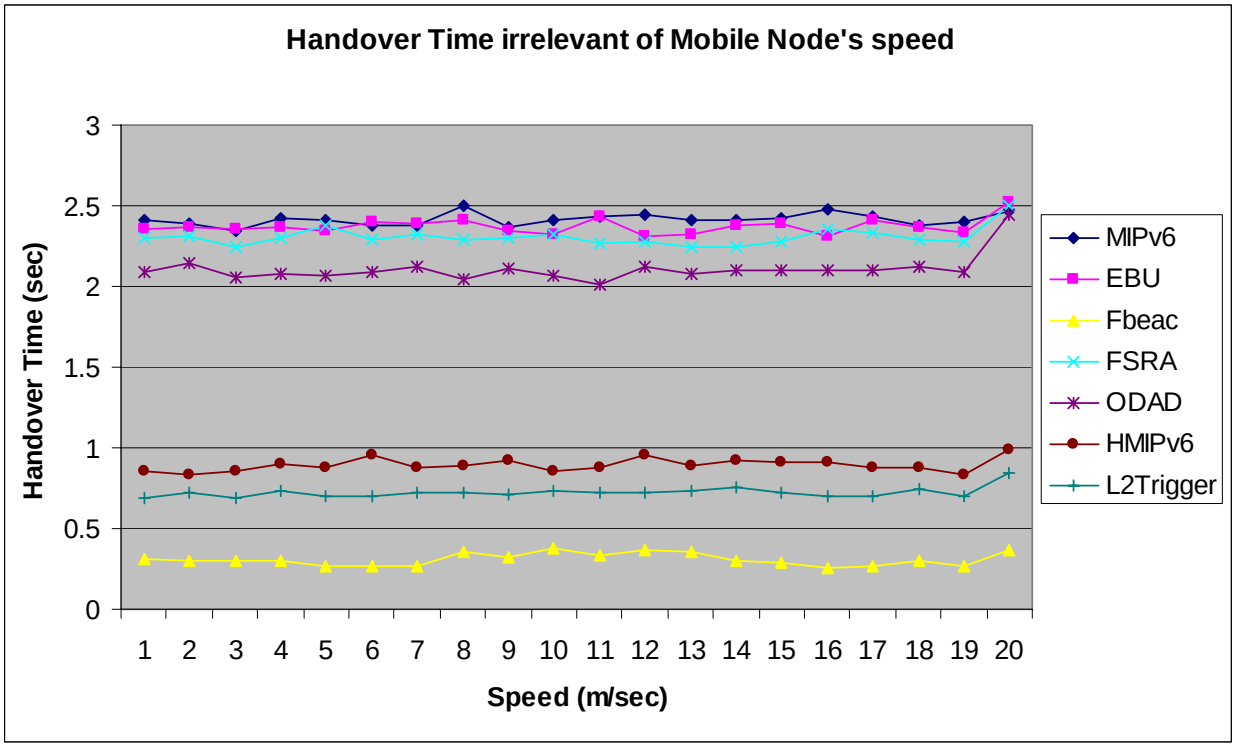
6. ΠΑΡΟΥΣΙΑΣΗ ΑΠΟΤΕΛΕΣΜΑΤΩΝ

6.1. Εξάρτηση από ταχύτητα

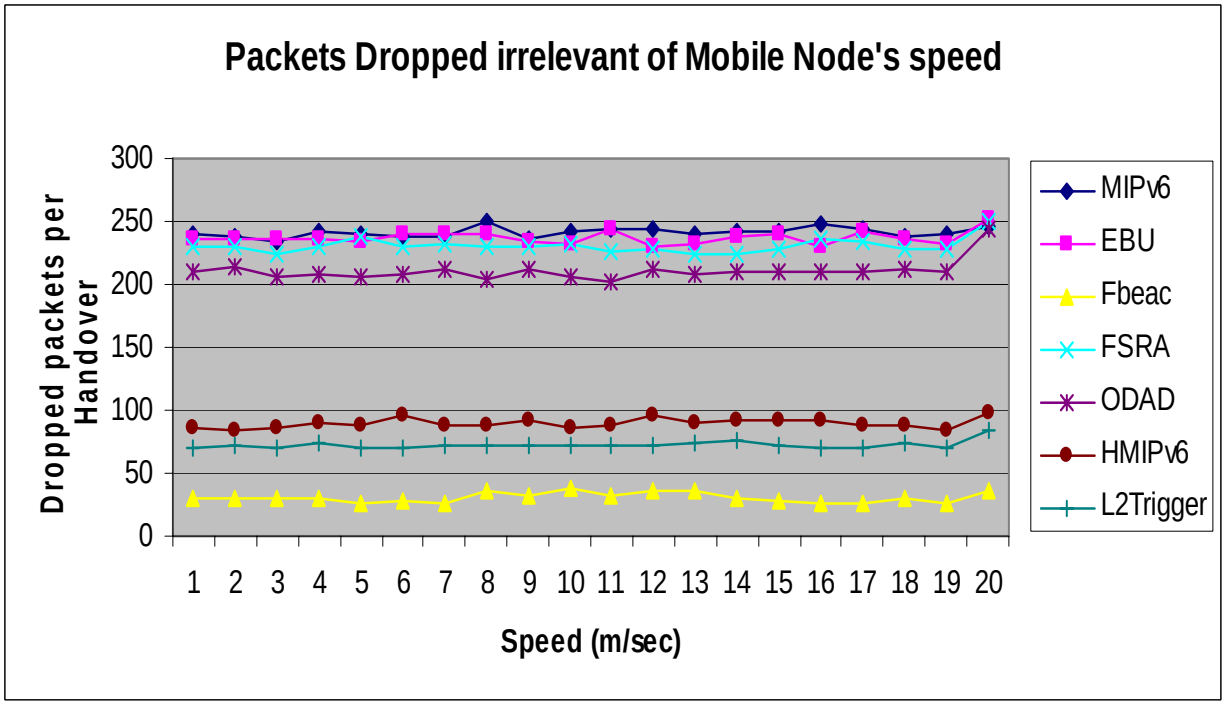
Θα εξετάσουμε αρχικά αν και κατά πόσο επηρεάζεται η handover καθυστέρηση και η απώλεια των πακέτων σε σχέση με την ταχύτητα του κινητού κόμβου. Τρέχουμε λοιπόν το μοντέλο που παρουσιάσαμε παραπάνω για κάθε μία από τις προαναφερθείσες τεχνικές, μεταβάλλοντας την ταχύτητα του κινητού κόμβου από 1 m/sec έως 20m/sec και καταγράφουμε τα αποτελέσματα. Προκειμένου να έχουμε μεγαλύτερη ακρίβεια, εκτελούμε τα πειράματα μας 10 φορές για κάθε τεχνική και για κάθε ταχύτητα, δίνοντας σαν είσοδο διαφορετικό seed για κάθε εκτέλεση. Καταλήγουμε να έχουμε λοιπόν 10*9=91 handovers για κάθε περίπτωση, αριθμός αρκετός ώστε να παρέχει την απαραίτητη αξιοπιστία.

Στις Εικόνες 21 και 22 παρουσιάζουμε τα αποτελέσματα μας. Στην Εικόνα 21 βλέπουμε ένα γράφημα που δείχνει τη σχέση μεταξύ της ταχύτητας του κινητού κόμβου και της handover καθυστέρησης, ενώ στην Εικόνα 22 το αντίστοιχο για την απώλεια πακέτων ανά handover.

Παρατηρώντας τα γραφήματα συμπεραίνουμε πως η ταχύτητα του κινητού κόμβου είναι ανεξάρτητη της handover καθυστέρησης για κάθε τεχνική. Κάτι τέτοιο δεν θα πρέπει να μας εκπλήσσει καθόλου αν αναλογιστούμε τις διαδικασίες που μεσολαβούν για την ολοκλήρωση ενός L3 handover. Ειδικότερα, προκειμένου ένας κόμβος να εγγραφεί σε κάποιο νέο δίκτυο, απαιτείται η ανταλλαγή μηνυμάτων μεταξύ του κινητού κόμβου και του αντίστοιχου δρομολογητή, τα οποία δεν επηρεάζονται από τόσο μικρές ταχύτητες όπως αυτές που αναπτύσσει ο κινητός κόμβος.



Εικόνα 21: Σχέση ταχύτητας MN και handover καθυστέρησης



Εικόνα 22: Σχέση ταχύτητας MN και χαμένων πακέτων

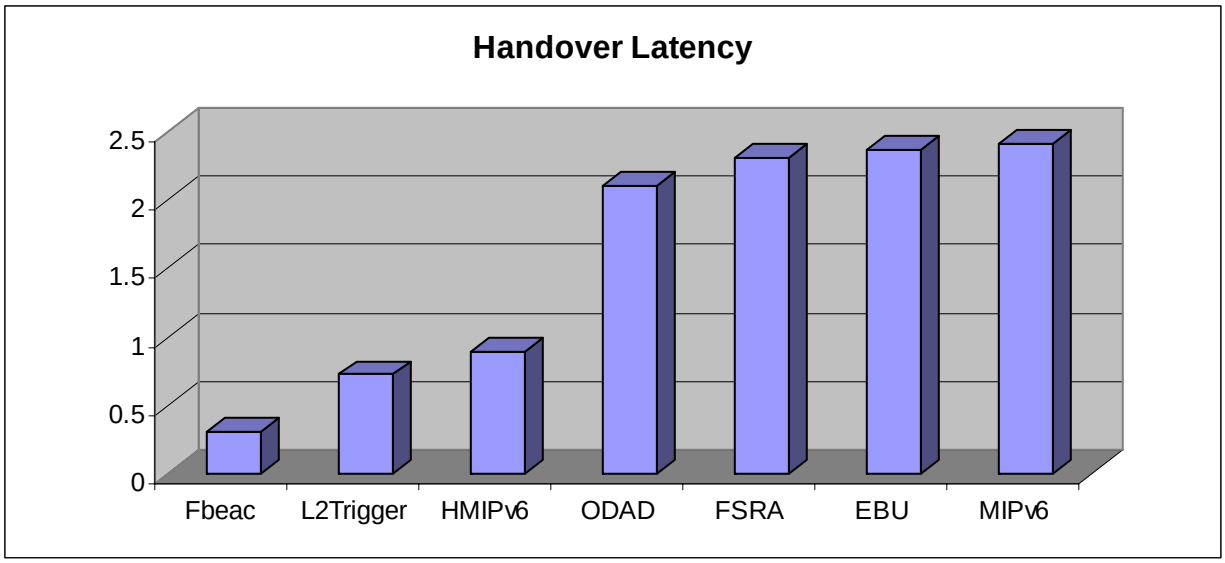
6.2. Βασικές επεκτάσεις

Με δεδομένο πια ότι η handover καθυστέρηση δεν εξαρτάται από την ταχύτητα του κινητού κόμβου, συνεχίζουμε τα πειράματα και προσπαθούμε να αξιολογήσουμε ανεξάρτητα κάθε μία από τις τεχνικές μας. Στα πειράματα που πραγματοποιήθηκαν η ταχύτητα του κόμβου ήταν σταθερή με 10 m/sec (36 km/h). Η συγκεκριμένη ταχύτητα επιλέχτηκε έτσι ώστε να ανταποκρίνεται και στην πραγματικότητα. Αν σκεφτούμε λοιπόν ένα όχημα που κινείται στην πόλη και διασχίζει διάφορα υποδίκτυα, κυψέλες και μικροκυψέλες, τότε αμέσως μπορούμε να δούμε τις ομοιότητες με το πείραμα μας.

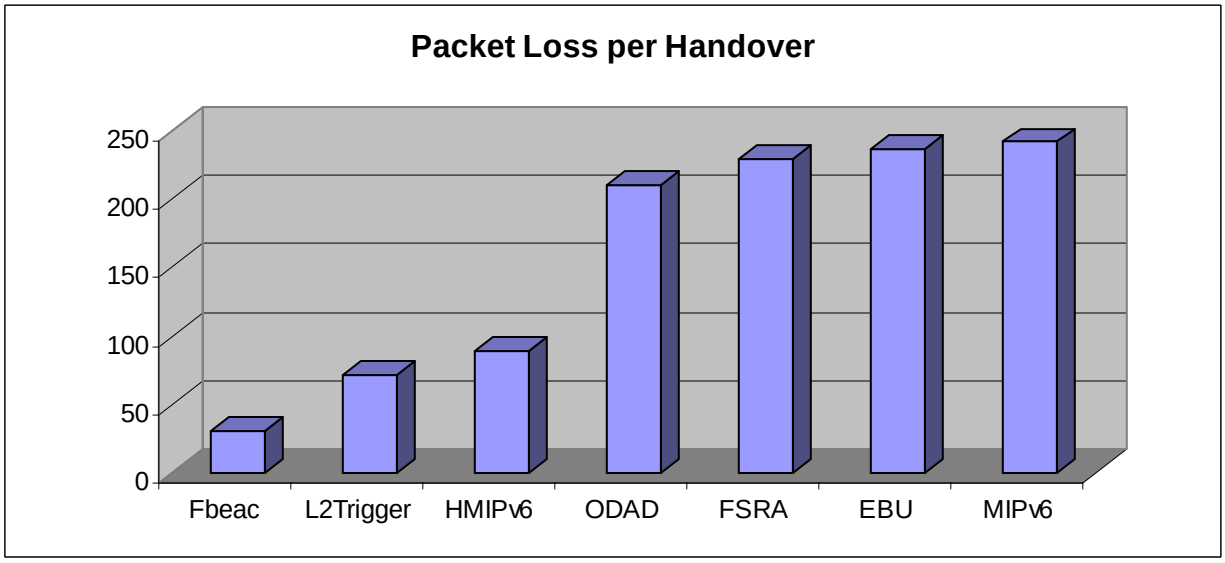
Όπως και πριν εκτελούμε 10 φορές το κάθε πείραμα με διαφορετικό seed κάθε φορά και καταγράφουμε τα αποτελέσματα μας. Ο παρακάτω πίνακας καταγράφει τα αποτελέσματα που πήραμε για την handover καθυστέρηση και την απώλεια πακέτων ανά handover.

Τεχνική	Avg (sec)	Std Dev	Min (sec)	Max (sec)	Packet loss
Fbeac	0.306906	0.039364	0.250366	0.376212	31
L2Trigger	0.722721	0.033074	0.690916	0.840925	72
HMIPv6	0.893267	0.040732	0.831776	0.984571	89
ODAD	2.107279	0.084238	2.01504	2.440599	210
FSRA	2.305996	0.057003	2.242407	2.498534	230
EBU	2.371481	0.050416	2.307597	2.521666	237
MIPv6	2.414414	0.038788	2.346704	2.500142	242

Παραθέτουμε και γραφικά τα παραπάνω αποτελέσματα. Στη συνέχεια θα μελετήσουμε την κάθε μία τεχνική ξεχωριστά.



Εικόνα 23: Handover καθυστέρηση για τις βασικές τεχνικές



Εικόνα 24: Απώλεια πακέτων ανά handover για τις βασικές τεχνικές

6.2.1. Optimistic Duplicate Address Detection

Το ODAD προσφέρει μια βελτίωση 310ms στην handover καθυστέρηση σε σχέση με το MIPv6 όπως βλέπουμε στον πίνακα μας. Θα αναμέναμε μεγαλύτερη βελτίωση δεδομένου του γρηγορότερου σχηματισμού CoA, απαλείφοντας τον χρόνο που χρειάζεται το DAD και είναι ένα δευτερόλεπτο εξ ορισμού. Ο λόγος που συμβαίνει αυτό είναι ο rendezvous time. Πιο συγκεκριμένα το ODAD χρησιμοποιείται όποτε ένα νέο πρόθεμα διαφημίζεται από το δρομολογητή, δηλαδή όταν λάβει ένα νέο RA με διαφορετική διεύθυνση δικτύου. Όταν όμως ο κινητός κόμβος κινείται προς ένα νέο υποδίκτυο, δεν είναι ικανός να γνωρίζει την ύπαρξη ενός του νέου αυτού δικτύου έως ότου τον ενημερώσει ο μηχανισμός ανίχνευσης κίνησης. Προκειμένου να ενεργοποιηθεί ο μηχανισμός αυτός πρέπει να μεσολαβήσουν (MaxConsecMissedRtrAdv+1)*MIPv6MaxRtrAdvInterval=2*1.5=3 δευτερόλεπτα στο σενάριο μας, χωρίς να ληφθεί κανένα RA πριν ο MN συνειδητοποιήσει πως μετακινήθηκε σε ένα νέο υποδίκτυο. Αυτός είναι αρκετός χρόνος για μη ODAD κινητούς κόμβους να λάβουν ένα νέο RA και να ολοκληρώσουν τις DAD διαδικασίες πριν ενεργοποιηθεί ο μηχανισμός ανίχνευσης κίνησης. Ο πίνακας και τα γραφήματα επιβεβαιώνουν αυτόν το συλλογισμό.
Μια αλλαγή που πιθανόν να βελτίωνε την απόδοση του ODAD θα ήταν η αλλαγή της τιμής του MaxConsecMissedRtrAdv σε 0. Με αυτό τον τρόπο θα έχουμε ενεργοποίηση του μηχανισμού ανίχνευσης μετακίνησης άμεσα από την στιγμή που δεν θα παραλαμβάνεται ένα RA μέσα στο αναμενόμενο χρονικό διάστημα και συνεπώς μείωση του rendezvous time.

6.2.2. Fast Solicited Router Advertisements

Όπως φαίνεται από τον παραπάνω πίνακα, η χρήση FSRA επιφέρει μια βελτίωση περίπου 90ms της handover καθυστέρησης σε σχέση με το MIPv6. Σύμφωνα με όσα σχολιάσαμε στην παράγραφο 3.3 η βελτίωση έπρεπε να ήταν της τάξης των

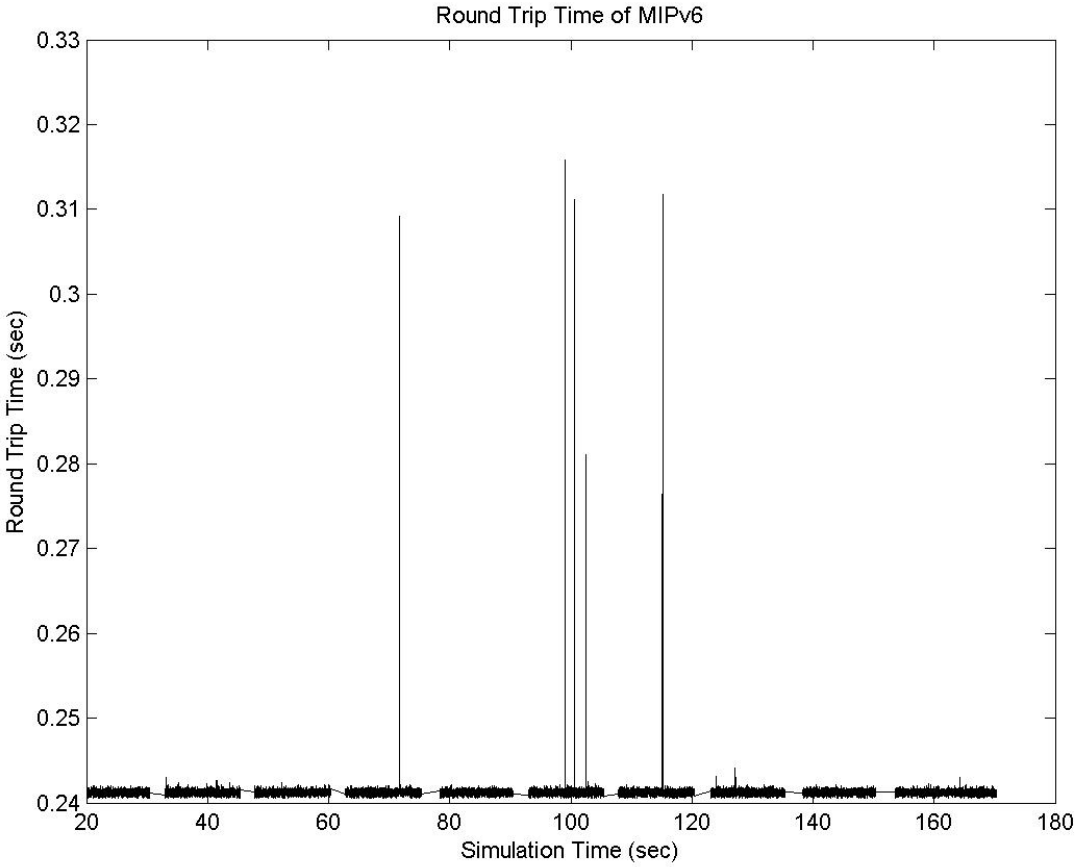
250ms[17], καθώς με την FSRA ο χρόνος αναμονής πριν από την απάντηση είναι 0. Το FSRA όμως λειτουργεί μόνο όταν ο κινητός κόμβος στείλει RS, δηλαδή μόνο όταν ο MN ανιχνεύσει τη μετακίνηση του σε ένα διαφορετικό υποδίκτυο. Συνεπώς η μικρή βελτίωση οφείλεται στο γεγονός ότι η αποστολή του RS γίνεται σε τυχαίο χρονικό διάστημα, δεδομένου ότι ο μηχανισμός ανίχνευσης μετακίνησης με νέο δίκτυο εξαρτάται από το τυχαίο διάστημα αποστολής unsolicited RA από τον δρομολογητή το οποίο κυμαίνεται από 1,5 έως 3 δευτερόλεπτα. Αυτό το διάστημα είναι μεγάλο, και σε αρκετές φορές, μεγαλύτερο από τον rendezvous time για το κλασσικό MIPv6 και για αυτό το λόγο παρατηρούμε μια αύξηση των χαμένων πακέτων ανά handover στην τεχνική FSRA.

6.2.3. Fast RA beacons

Παρατηρώντας την handover καθυστέρηση και την απώλεια πακέτων για την τεχνική των Fast RA beacons στον πίνακα μας, γίνεται ξεκάθαρο ότι η μείωση του διαστήματος μεταξύ unsolicited RA επιφέρει αξιοσημείωτα αποτελέσματα (1,8 δευτερόλεπτα!). Όπως εξηγήσαμε και παραπάνω η αποστολή ενός unsolicited RA θα προκαλέσει την έναρξη του μηχανισμού ανίχνευσης μετακίνησης. Στέλνοντας λοιπόν RA κατά μέσο όρο κάθε 50ms[18] το Layer 3 μπορεί να ανιχνεύσει την μετακίνηση πολύ γρήγορα και συνεπώς να αρχίσει τις επικείμενες διαδικασίες για την ανάκτηση μιας νέας CoA.
Η αποστολή RA κάθε 50ms, σημαίνει πως κάθε δευτερόλεπτο υπάρχουν 20 RA στο μέσο, 25 φορές πιο πολλά RA σε σχέση με τα 0.8 RA που στέλνονται υπό κανονικές συνθήκες. Κάτι τέτοιο μας υποψιάζει για αυξημένες συγκρούσεις μεταξύ των ping και RA πακέτων. Εκτελώντας επιπλέον πειράματα, καταγράφουμε τον Round Trip Time των ping πακέτων για την περίπτωση του απλού MIPv6 και των Fast RA beacons. Στην Εικόνα 25 βλέπουμε το Round Trip Time για την εκτέλεση ενός τυχαίου παραδείγματος του μοντέλου μας χρησιμοποιώντας μόνο την βασική MIPv6 τεχνική

[17] ο μέσος όρος του 0 και MAX_RA_DELAY_TIME
[18] Ο μέσος όρος του MIPv6MinRtrAdvInterval=30ms και MIPv6MaxRtrAdvInterval=50ms

και στην Εικόνα 26 το αντίστοιχο κάνοντας χρήση της Fast RA beacons τεχνικής αυτή τη φορά.

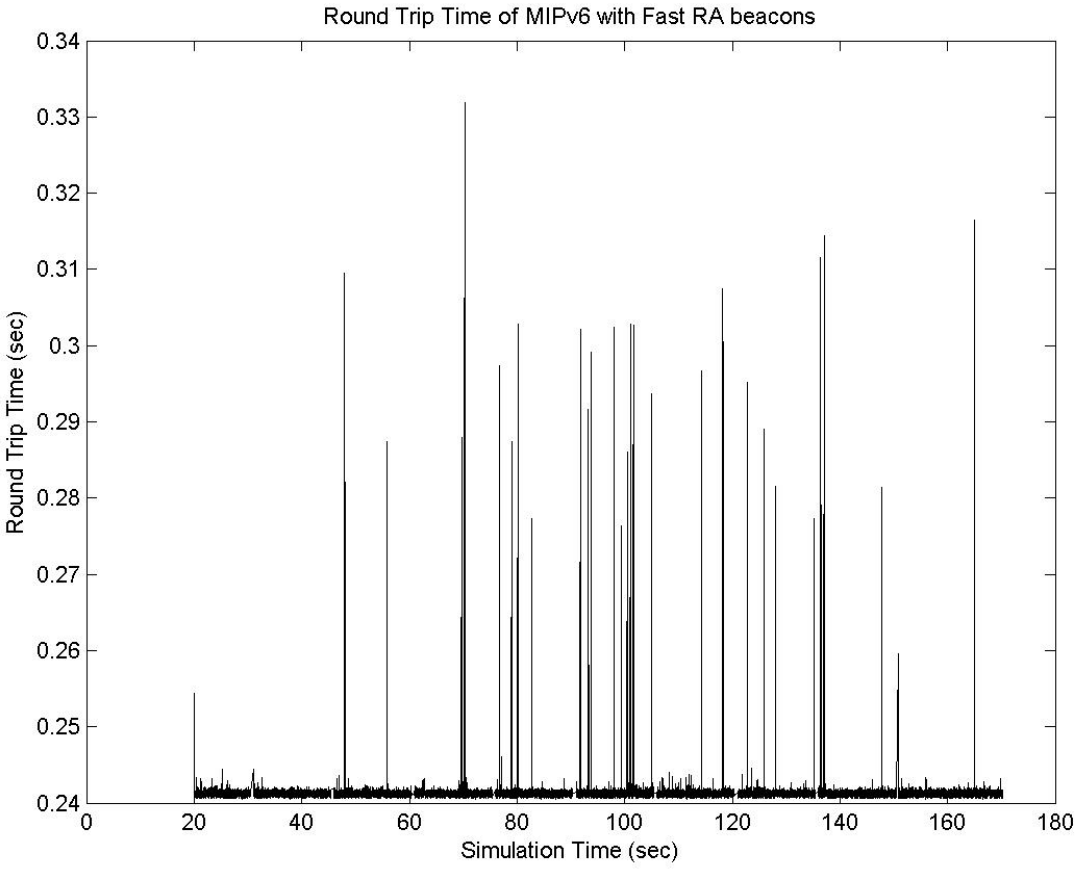


Εικόνα 25: Round Trip time για το κλασσικό MIPv6

Παρατηρούμε στην Εικόνα 25 πως η κατανομή είναι ομοιόμορφή, με ελάχιστες διακυμάνσεις και πως γενικά ο RTT είναι σταθερός. Οι ελάχιστες ακμές στο γράφημα οφείλονται πιθανότατα στις μη-ιδανικές προδιαγραφές του Φυσικού Επιπέδου που εξομοιώνεται στο μοντέλο μας.

Εξετάζοντας την περίπτωση των Fast RA beacons τώρα, βλέπουμε πως η μείωση του διαστήματος μεταξύ RA από 1-1,5sec σε 30-70ms επιφέρει μεγάλη αύξηση στον αριθμό των ακμών στο γράφημα μας. Αυτό οφείλεται στη σύγκρουση των RA με τα ping request ή acknowledgement πακέτα. Αυτό επιβεβαιώνει την αρχική υπόθεση μας ότι η πιθανότητα συγκρούσεων μεταξύ πακέτων έχει αυξηθεί λόγο της αύξησης της συχνότητας των RA.

Μπορούμε να πούμε συμπερασματικά πως η τεχνική των Fast RA beacons δεν ενδείκνυται για περιοχές με μεγάλη κίνηση και πολλούς κινητούς χρήστες καθώς οι συγκρούσεις θα είναι αυξημένες, όπως επίσης και δεν ενδείκνυται για μεταφορά streaming multimedia περιεχομένου καθώς οι απαιτήσεις στην διακύμανση πρέπει να είναι σταθερές και σαφώς ορισμένες, κάτι που δεν είναι δυνατό.



Εικόνα 26: Round Trip time για MIPv6 με Fast RA beacons

6.2.4. Early Binding Updates

Σύμφωνα με τα όσα περιγράψαμε στην παράγραφο 3.7 η χρήση της τεχνικής των Early Binding Updates θα επέφερε βελτίωση της τάξης του ενός RTT από τον MN

προς τον CN, σε σχέση με το απλό MIPv6. Τα αποτελέσματα μας δεν συμβαδίζουν ακριβώς με αυτή την άποψη, αλλά αντιθέτως δίνουν μια βελτίωση των 33ms συγκριτικά με τα 240ms που θα έπρεπε να αναμέναμε. Ο λόγος που συμβαίνει αυτό είναι πολύ απλός αρκεί να ξαναδούμε πως δουλεύουν τα EBU.

Ένας κινητός κόμβος πρέπει μόλις αντιληφθεί ένα νέο υποδίκτυο ή ανά τακτά χρονικά διαστήματα να εκτελεί το Home-Address Test με στόχο την μείωση της καθυστέρησης καταχώρησης της NCoA. Σύμφωνα με την προτεινόμενη τεχνική λοιπόν ένα Home-Address Test πρέπει να πραγματοποιείται τουλάχιστον κάθε 3,5 λεπτά ή όταν ανιχνευτεί ένα νέο υποδίκτυο. Οπότε στο σενάριο μας τα EBU αποτυγχάνουν σε κάθε περίπτωση. Η ανίχνευση ενός καινούριου υποδικτύου και άρα η εκκίνηση του Home-Address Test εξαρτάται από τον χρόνο ενεργοποίησης του μηχανισμού ανίχνευσης μετακίνησης, ο οποίος κυμαίνεται από 1,5-3 δευτερόλεπτα και συνεπώς δεν προσφέρει κανένα ουσιαστικό πλεονέκτημα στην EBU τεχνική συγκριτικά με το MIPv6. Από την άλλη η περίοδος των 3,5 λεπτών είναι πολύ μεγάλη για το σενάριο μας που πραγματοποιείται handover κάθε 15 δευτερόλεπτα περίπου.

Μία λύση που θα μπορούσε να βελτιώσει την απόδοση της EBU τεχνικής σε δίκτυα μικροκυψέλων, όπως το δικό μας, θα μπορούσε να είναι η πραγματοποίηση πολύ πιο συχνών Home-Address Tests με αντάλλαγμα το επιπλέον overhead στο δίκτυο.

6.2.5. L2 Triggers

Η απόδοση της L2 Trigger τεχνικής είναι επίσης αξιοσημείωτη. Η ενημέρωση του Layer 3 για ενδεχόμενο handover και συνεπώς η αποστολή RS από τον κινητό κόμβο, μπορεί να εξοικονομήσει 1,7 δευτερόλεπτα από την συνολική handover καθυστέρηση όπως φαίνεται από τα πειράματα μας.

Η L2 trigger τεχνική εκτός του ότι δρα καταλυτικά στην μείωση της handover καθυστέρησης, έχει και ένα ακόμα μεγάλο πλεονέκτημα. Η εισαγωγή της και η λειτουργία της από έναν κινητό κόμβο είναι πάρα πολύ απλή και οικονομικά φτηνή.

Απαιτεί μόνο μια αναβάθμιση της MIPv6 στοίβας, έτσι ώστε να προωθεί το ήδη υλοποιημένο και εν λειτουργία "L2 Link Up" μήνυμα στο ανώτερο επίπεδο.
Τα μηνύματα αυτά όμως είναι πιθανόν, όπως είδαμε και στην παράγραφο 3.2, να είναι παραπλανητικά. Αυτό μπορεί να συμβεί στην περίπτωση που ο κινητός κόμβος μετακινείται μεταξύ διαφόρων Access Points τα οποία όμως είναι συνδεδεμένα στην ίδια διεπαφή του υποδικτύου και συνεπώς δε επίκειται κάποιο L3 handover. Τότε ο κινητός κόμβος θα στέλνει περιττά RS αυξάνοντας την κίνηση στο δίκτυο. Ακόμα και έτσι όμως, και δεδομένων διαφόρων λύσεων που μπορούν να λύσουν εν μέρει το παραπάνω πρόβλημα, το όφελος της τεχνικής υπερνικά την οποιαδήποτε περιττή κίνηση στο δίκτυο.

6.2.6. Hierarchical Mobile IPv6

Σύμφωνα με τα όσα είδαμε στην παράγραφο 4.2, ο MAP παίζει τον ρόλο του HA στο τοπικό δίκτυο. Συνεπώς ο κινητός κόμβος στέλνει τις BU στον τοπικό MAP και όχι στον HA και στους CNs που βρίσκονται πιο μακριά. Αυτό έχει σαν αποτέλεσμα την μείωση της handover καθυστέρησης κατά ελάχιστο 1,5 round-trip time σε σχέση με το MIPv6, επειδή δεν απαιτείται πια η return routability διαδικασία για κάθε CN. Στο μοντέλο μας λοιπόν θα αναμέναμε μείωση της καθυστέρησης κατά τουλάχιστον 1,5*0,24=0.36 δευτερόλεπτα. Η μετρούμενη βελτίωση όμως είναι κατά πολύ μεγαλύτερη και αγγίζει τα 1,5 δευτερόλεπτα, 190ms πιο αργή από την L2 Trigger τεχνική.
Αναλύοντας τις μετρήσεις μας για το πρώτο handover[19], δηλαδή το handover κατά το οποίο ο κινητός κόμβος εισέρχεται στην περιοχή του MAP, παρατηρούμε πως η handover καθυστέρηση είναι κατά μέσο όρο 1,8 δευτερόλεπτα. Σύμφωνα με το HMIPv6 ο κινητός κόμβος κατά την πρώτη επαφή του με τον MAP πρέπει να συνδέσει την δική του CoA με μία RCoA του υποδικτύου του MAP και στη συνέχεια αυτή η RCoA να διαφημιστεί προς τους CN και HA.

[19] Το πρώτο handover δεν λαμβάνεται υπόψη στους υπολογισμούς μας καθώς θέλαμε να μελετήσουμε την συμπεριφορά της κάθε τεχνικής σε ένα καθαρό δίκτυο, μετά τις αρχικοποιήσεις

Εφόσον όμως ο κινητός κόμβος κινείται στο domain του MAP ο MN απλά γνωστοποιεί την NCoA στον MAP. Η RCoA δεν χρειάζεται να αλλάζει, η κίνηση του είναι διαφανής προς τον CN και δεν απαιτούνται επιπλέον bindings. Για αυτό ακριβώς το λόγο η handover καθυστέρηση και ο αριθμός των χαμένων πακέτων είναι ακόμα μικρότερα.
Η Hierarchical Mobile IPv6 τεχνική λοιπόν καταφέρνει να εξασφαλίσει πολύ καλούς handover χρόνους, ισάξιους σχεδόν με την L2 trigger, αλλά χάνει στην πολυπλοκότητα. Για την λειτουργία του HMIPv6 απαιτούνται αρκετές αλλαγές στην στοίβα του MIPv6, όπως η δημιουργία νέων μηνυμάτων (LBU), καθώς και χειροκίνητο προσδιορισμό εκείνων των δρομολογητών που θα δρουν σαν MAP.

6.3. Συνδυασμοί βασικών επεκτάσεων

Αφού είδαμε και σχολιάσαμε τις αποδόσεις, τα πλεονεκτήματα και τα μειονεκτήματα της κάθε τεχνικής ξεχωριστά, τώρα θα προσπαθήσουμε να βρούμε εκείνους τους συνδυασμούς που αποτελούν την καλύτερη λύση για την ελαχιστοποίηση της handover καθυστέρησης και της απώλειας πακέτων, έχοντας πάντα υπόψη την πολυπλοκότητα και το κόστος της κάθε τεχνικής.
Επαναλαμβάνουμε λοιπόν το προηγούμενο πείραμα, με τα ίδια τεχνικά χαρακτηριστικά[20], για κάθε δυνατό συνδυασμό των παραπάνω τεχνικών. Συνολικά έχουμε 48 δυνατούς συνδυασμούς και στον παρακάτω πίνακα παραθέτουμε τα αποτελέσματα μας. Στον παρακάτω πίνακα παρουσιάζονται όλες οι εξεταζόμενες τεχνικές διατεταγμένες κατά φθίνουσα σειρά ως προς την handover καθυστέρηση, ενώ αναφέρουμε επίσης τον αριθμό των χαμένων πακέτων ανά handover.

Technique	Handover Latency	Packet loss per handover
MIPv6	2.4140	244
EBU	2.3778	237.7
FSRA	2.3123	231.2

[20] Κινητός κόμβος διατρέχει την τοπολογία μας με 10m/s στέλνοντας ping πακέτα κάθε 10ms

EBU FSRA	2.3080	230.8
EBU ODAD	2.1132	211.3
FSRA ODAD	2.1117	211.1
EBU FSRA ODAD	2.1113	211.1
ODAD	2.1097	210.9
HMIPv6 ODAD	0.9416	94.1
FSRA HMIPv6	0.9358	93.5
FSRA HMIPv6 ODAD	0.9309	93
HMIPv6	0.8939	89.3
L2Trig FSRA	0.7267	72.6
L2Trig EBU	0.7241	72.4
L2Trig	0.7241	72.4
L2Trig EBU FSRA	0.7215	72.1
L2Trig EBU ODAD	0.6350	63.5
L2Trig EBU FSRA ODAD	0.6336	63.3
L2Trig FSRA ODAD	0.6323	63.2
L2Trig HMIPv6 ODAD	0.6308	63
L2Trig ODAD	0.6305	63
L2Trig FSRA HMIPv6	0.6303	63
L2Trig FSRA HMIPv6 ODAD	0.6281	62.8
L2Trig HMIPv6	0.6200	62
Fbeac FSRA HMIPv6 ODAD	0.3209	32
Fbeac HMIPv6 ODAD	0.3201	32
L2Trig Fbeac HMIPv6 ODAD	0.3199	31.9
L2Trig Fbeac FSRA HMIPv6 ODAD	0.3163	31.6
L2Trig EBU Fbeac	0.3154	31.5
Fbeac HMIPv6	0.3153	31.5
EBU Fbeac FSRA	0.3150	31.5
EBU Fbeac	0.3136	31.3
L2Trig Fbeac	0.3131	31.3
L2Trig Fbeac FSRA	0.3130	31.3
Fbeac SRA HMIPv6	0.3128	31.2
L2Trig Fbeac HMIPv6	0.3126	31.2
L2Trig Fbeac FSRA HMIPv6	0.3122	31.2
Fbeac FSRA	0.3111	31.1
L2Trig Fbeac FSRA ODAD	0.3109	31
Fbeac	0.3108	31
L2Trig EBU Fbeac FSRA ODAD	0.3105	31
L2Trig Fbeac ODAD	0.3101	31
Fbeac ODAD	0.3095	30.9
EBU Fbeac FSRA ODAD	0.3092	30.9
L2Trig EBU Fbeac FSRA	0.3073	30.7
Fbeac FSRA ODAD	0.3071	30.7
EBU Fbeac ODAD	0.3064	30.6
L2Trig EBU Fbeac ODAD	0.3055	30.5

Το πρώτο πράγμα που μπορούμε να σχολιάσουμε κοιτώντας με μια γρήγορη ματιά τα αποτελέσματα είναι πως η συνολική μείωση της handover καθυστέρησης δεν ισούται απαραίτητα με το άθροισμα των βελτιώσεων της κάθε τεχνικής, ενώ υπάρχουν και περιπτώσεις που ο συνδυασμός κάποιων τεχνικών είναι πιο αργός από την κάθε μία ξεχωριστά! Παραδείγματος χάριν ο συνδυασμός της FSRA και της HMIPv6 τεχνικής έχει χειρότερη επίδοση από ότι η τεχνική HMIPv6 από μόνη της, ενώ ο συνδυασμός L2 Triggers, EBU, ODAD έχει ακριβώς την ίδια απόδοση με τον συνδυασμό L2 Trigger, ODAD κάνοντας έτσι την τεχνική EBU να δείχνει περιττή.

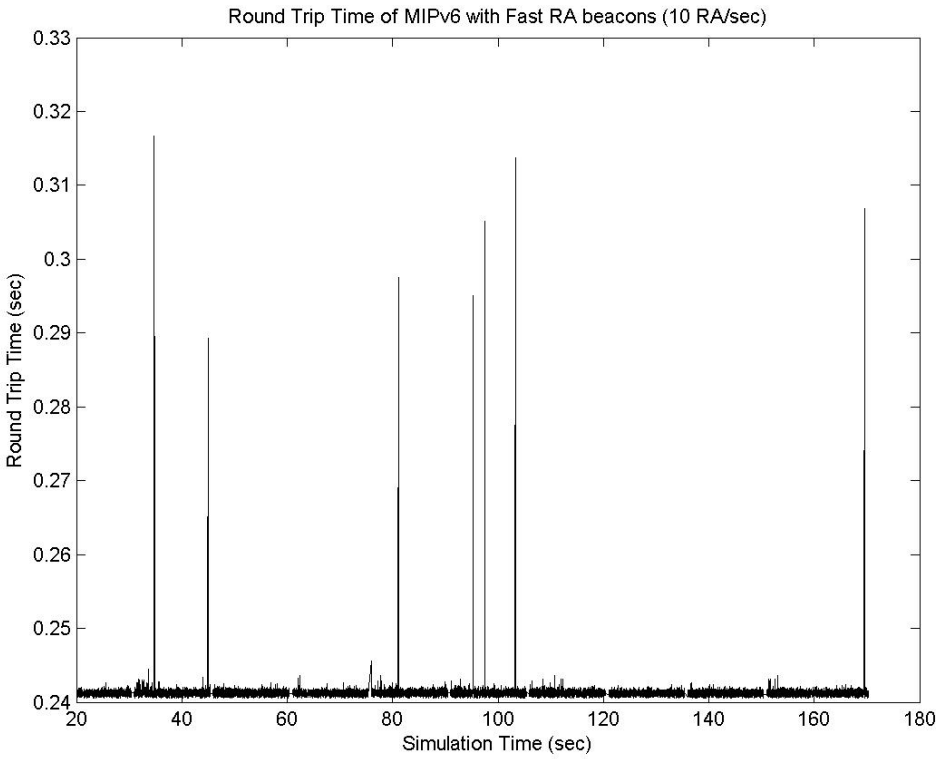
Συνεχίζοντας θα σχολιάσουμε την συμπεριφορά της L2 Trigger τεχνικής καθώς αυτή συνδυάζεται με άλλες τεχνικές. Ο χρόνος των 0,72 δευτερολέπτων που επιτυγχάνει από μόνη της, φαίνεται να είναι πολύ κοντά στο κάτω όριο της απόδοσης όλων των συνδυασμών[21] της. Ακόμα και στην καλύτερη περίπτωση που χρησιμοποιείται μαζί με την HMIPv6, η βελτίωση είναι μικρότερη από 100ms. Ο συνδυασμός των τεχνικών L2 Trigger και ODAD φαίνεται πάλι να επιβεβαιώνει τα προηγούμενα καθώς η βελτίωση με τη χρήση του ODAD ανέρχεται μόλις στα 90ms, πολύ μικρότερη από το θεωρητικό 1 δευτερόλεπτο που χρειάζεται το ODAD εξ' ορισμού. Από την άλλη μεριά θα ήταν παράλογο να περιμέναμε μια τέτοια απόδοση καθώς τότε η καθυστέρηση θα ήταν αρνητική! Συνοψίζοντας λοιπόν, η τεχνική L2 Trigger είναι από τις πιο σημαντικές στην έρευνα για μείωση του handover χρόνου για τους δύο παρακάτω λόγους. Αφενός η υλοποίηση και εφαρμογή της στον κινητό κόμβο είναι πάρα πολύ απλή και με σχεδόν μηδαμινό κόστος, ενώ αφετέρου είναι ικανή να πετύχει αξιοσημείωτες επιδόσεις ακόμα και όταν χρησιμοποιείται μόνη της.

Επιδιώκοντας μια ακόμα καλύτερη επίδοση καταλήγουμε στην χρήση των Fast RA beacons. Βλέπουμε πως η χρήση τους, ακόμα και χωρίς τον συνδυασμό τους με κάποια άλλη τεχνική, έχει σημαντικότατα αποτελέσματα. Σύμφωνα με τις μετρήσεις μας λοιπόν θέτουμε σαν κάτω όριο της handover καθυστέρησης τα 300ms. Παρατηρώντας τα 20 καλύτερα αποτελέσματα μας στον πίνακα μας συμπεραίνουμε πως μια τέτοια απόδοση μπορεί να προσεγγιστεί μόνο με την χρήση των Fast Beacons. Ανατρέχοντας στην παράγραφο 6.2.3 όμως, είδαμε πως η χρήση των Fast beacons έχει σαν παρενέργεια την αυξημένη διακύμανση του RTT των πακέτων λόγω των συγκρούσεων μεταξύ των RA και των πακέτων. Καταλήγουμε λοιπόν σε

[21] Εξαιρώντας τον συνδυασμό της με την τεχνική Fbeac που θα ασχοληθούμε παρακάτω

ένα δίλημμα. Να ανεχτούμε το αυξημένο jitter για χάρη του πολύ καλού handover χρόνου, ή να αρκεστούμε στον επόμενο καλύτερο χρόνο που δεν κάνει χρήση της τεχνικής Fast Beacons και είναι ο διπλάσιος;

Ένα τέτοιο δίλημμα είναι πολύ κρίσιμο που εξαρτάται από πολλούς παράγοντες, όπως το είδος της κίνησης μέσα στο κανάλι, το διαθέσιμο εύρος ζώνης κ.α. Συνεπώς είναι αρκετά δύσκολο να καταλήξουμε κάπου αμέσως. Επιστρέφουμε λοιπόν στα πειράματα μας και επαναλαμβάνουμε το πείραμα των Fast Beacons αλλά με διαφορετικές παραμέτρους. Καταγράφουμε λοιπόν πάλι τον RTT για τα Ping πακέτα μας, αυτή τη φόρα όμως μεταβάλουμε τις παραμέτρους MIPv6MinRtrAdvInterval και MIPv6MaxRtrAdvInterval.

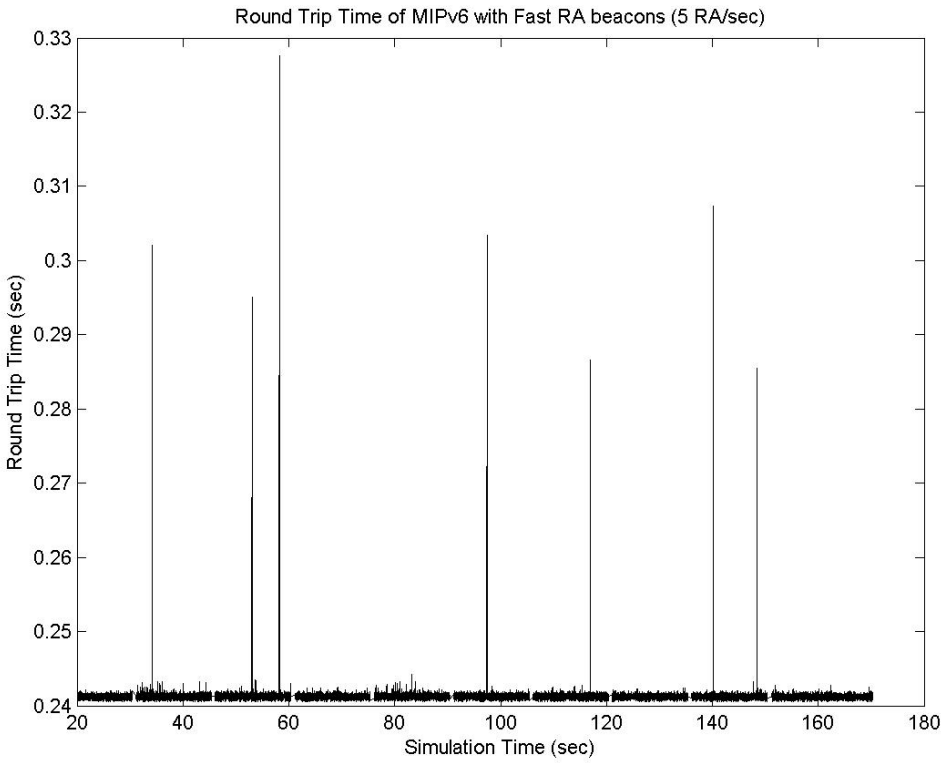


Εικόνα 27: Round Trip time για MIPv6 με 10 RA ανά δευτερόλεπτο

Στην Εικόνα 27 φαίνονται τα αποτελέσματα μας αν χρησιμοποιήσουμε την Fast Beacon τεχνική με τις διπλάσιες των ελάχιστων τιμών για τις μεταβλητές MIPv6MinRtrAdvInterval και MIPv6MaxRtrAdvInterval. Έτσι τώρα έχουμε 10 RA κατά μέσο όρο ανά δευτερόλεπτο. Παρατηρούμε πως ο αριθμός των ακμών στο γράφημα

έχουν μειωθεί πάρα πολύ σχετικά με αυτές της Εικόνας 26, δηλαδή όταν στέλνουμε 20 RA ανα δευτερόλεπτο. Επιπλέον παρατηρούμε πως το γράφημα προσεγγίζει πολύ καλά το γράφημα της Εικόνας 25 που αφορά το κλασσικό MIPv6. Καταφέραμε λοιπόν, μειώνοντας τον αριθμό των RA ανά δευτερόλεπτο στο μισό να μειώσουμε κατά πολύ το jitter και κατά συνέπεια και την συνολική απόδοση του συστήματος μας, καθώς πια δεν είμαστε δέσμιοι του είδους της κίνησης.

Το ερώτημά όμως που τίθεται, είναι ποια η handover συμπεριφορά του μοντέλου αυτού. Αυξήθηκε πολύ η handover καθυστέρηση; Οι μετρήσεις μας είναι πολύ ενθαρρυντικές. Η μέση handover καθυστέρηση μετρήθηκε στα 0.336 δευτερόλεπτα, δηλαδή 26ms περισσότερα από του Fast Beacon με τα 20 RA/sec, πολύ μικρό overhead αν αναλογιστούμε τι κερδίσαμε.



Εικόνα 28: Round Trip time για MIPv6 με 5 RA ανά δευτερόλεπτο

Δεδομένων αυτών των ενθαρρυντικών αποτελεσμάτων κάνουμε άλλο ένα πείραμα μειώνοντας ακόμα πιο πολύ τον ρυθμό μετάδοσης RA, στα 5 RA/sec. Η Εικόνα 28 δείχνει τα αποτελέσματα μας. Σε αυτή την περίπτωση ενώ η διακύμανση είναι παρόμοια με το προηγούμενο πείραμα, η handover καθυστέρηση που μετρήθηκε

είναι μεγαλύτερη κατά 130ms, αγγίζοντας τα 0,46 δευτερόλεπτα. Μπορούμε λοιπόν να αποκλείσουμε αυτό το μοντέλο, καθώς συγκριτικά με το προηγούμενο δεν μας προσφέρει τίποτα παραπάνω, ενώ αντίθετα έχει χειρότερη απόδοση.

Σύμφωνα με αυτά, προκειμένου να έχουμε την καλύτερη handover καθυστέρηση σε συνδυασμό με το μικρότερο jitter, καταλήγουμε στη χρήση της Fast Beacons τεχνικής, παραλλαγμένη έτσι ώστε να μεταδίδονται 10 RA το δευτερόλεπτο αντί για 20 που είναι και το άνω όριο που επιτρέπει το [17]. Η τεχνική, έχει επίσης μηδαμινό κόστος υλοποίησης καθώς δεν εισάγονται νέα στοιχεία στη στοίβα του MIPv6, αλλά αρκεί μία ρύθμιση στους δρομολογητές, κάνοντας την έτσι ακόμα πιο ελκυστική.

7. ΣΥΜΠΕΡΑΣΜΑΤΑ

Στο εγγύς μέλλον η μορφή των ασύρματων δικτύων θα είναι τελείως διαφορετική. Η αύξηση των κινητών χρηστών, καθώς και οι ολοένα μεγαλύτερες ανάγκες τους θα ωθήσουν στην χρήση micro-κυψέλων και pico-κυψέλων με αποτέλεσμα την μείωση της περιοχής κάλυψης σε οικοδομήματα, ακόμα και σε δωμάτια. Για ένα χρήστη που αλλάζει συνεχώς σημεία σύνδεσης στο δίκτυο τα επαναλαμβανόμενα handover θα επηρεάζουν και θα υποβαθμίζουν τα χαρακτηριστικά των κλήσεων του.

Στόχος αυτής της διπλωματικής ήταν η αξιολόγηση και ανεύρεση εκείνων των τεχνικών που σε ένα δίκτυο IPv6 θα μειώνουν την handover καθυστέρηση στο ελάχιστο.

Με χρήση εξομοιωτή μετρήσαμε την απόδοση διάφορων τεχνικών και καταλήξαμε πως για να επιτύχουμε την καλύτερη δυνατή απόδοση, καταλυτικό ρόλο παίζει η τεχνική των Fast RA Beacons. Είναι ικανή να μειώσει την handover καθυστέρηση στα 300ms από τα 2,5 δευτερόλεπτα του κλασσικού MIPv6. Η χρήση της σε συνδυασμό με οποιαδήποτε άλλη τεχνική επιφέρει εξίσου σημαντικά αποτελέσματα, αλλά όχι τόσο ώστε να δικαιολογεί το επιπλέον κόστος και πολυπλοκότητα που εισάγεται. Επιπλέον δείξαμε πως το ανεπιθύμητο jitter που εισάγει η τεχνική των Fast Beacons μπορεί να αντιμετωπιστεί μειώνοντας τον αριθμό των RA που στέλνουν οι δρομολογητές το δευτερόλεπτο στο μισό, χωρίς να χάνουμε σημαντικά σε απόδοση.

Η επόμενη πολύ σημαντική τεχνική είναι αυτή των L2 Triggers. Όπως αναφέραμε στην παράγραφο 6.2 η απόδοση των περισσοτέρων τεχνικών εξαρτάται από την άμεση ανίχνευση μετακίνησης σε ένα νέο υποδίκτυο. Σε αυτό ακριβώς συμβάλει η τεχνική των L2 Triggers βελτιώνοντας κατά πολύ την συνολική απόδοση και μειώνοντας αισθητά την handover καθυστέρηση. Χαρακτηρίζουμε αυτή την τεχνική σαν μία πολύ καλή επιλογή δεδομένου του μηδαμινού κόστους εφαρμογής της και της σημαντικής απόδοσης της.

Η HMIPv6 τεχνική δεν καταφέρνει να μας πείσει, μιας και αφενός οι επιδόσεις που καταφέρνει δεν την κατατάσσουν σαν βέλτιστη επιλογή, αφετέρου η δημιουργία, εγκατάσταση και διαχείριση ενός ιεραρχικά δομημένου δικτύου είναι μια εν γένει

δύσκολη διαδικασία. Η επιλογή εκείνου του δρομολογητή που θα ενεργεί ως MAP αποτελεί ένα περίπλοκο πρόβλημα καθώς οι LMA αλγόριθμοι επιλογής δεν είναι ντετερμενιστικοί και δεν εγγυούνται την καλύτερη απόδοση. Επίσης απαιτούνται μάλλον ακριβές αλλαγές στην στοίβα του MIPv6 πρωτοκόλλου καθώς και η χειροκίνητη ρύθμιση των ενδιάμεσων δρομολογητών. Έτσι λοιπόν η HMIPv6 τεχνική αν και πολλά υποσχόμενη, στο μοντέλο μας δεν σημείωσε αξιοσημείωτες επιδόσεις.

ΠΑΡΑΡΤΗΜΑ

OMNet++ και IPv6Suite

Παραθέτουμε όλα τα απαραίτητα αρχεία και ρυθμίσεις που χρησιμοποιήθηκαν για την αρχικοποίηση και λειτουργία των πειραμάτων μας.

Το αρχείο διαμόρφωσης της τοπολογίας (L3HO.ned)

```
import
    "Router6",
    "UDPNode",
    "WorldProcessor",
    "WirelessAccessPoint",
    "WirelessMobileNode";



channel MIPv6SimpleInternetCable
      delay 1e-1;
    datarate 10e9;
endchannel

channel MIPv6SimpleIntranetCable
    delay 1.5e-6; // propagation delay for 30 meter link
    datarate 100e6;
endchannel

channel LMACable
//Large delay means large map domain/ small means small map domain
    delay 2e-2;
//    delay 2e-3;
//    delay 5e-2;
    datarate 1e9;
endchannel

module L3HO
    submodules:
        worldProcessor: WorldProcessor;
            display: "p=672,31;i=bwgen_s";
        client1: MobileNode;
            parameters:
                IPForward = false;
        gatesizes:
            wlin[1],
            wlout[1];
            display: "p=32,334;i=laptop3";
        server: UDPNode;
            parameters:
                IPForward = false;
```

```
            gatesizes:
                in[1],
                out[1];
            display: "p=407,41;i=pc";
        ar: Router6;
            gatesizes:
                in[11],
                out[11];
            display: "p=480,208;i=router";
        ap1: AccessPoint;
            gatesizes:
                in[1],
                out[1];
            display: "p=250,286;i=switch1_s";
        ap2: AccessPoint;
            gatesizes:
                in[1],
                out[1];
            display: "p=400,286;i=switch1_s";
        ap3: AccessPoint;
            gatesizes:
                in[1],
                out[1];
            display: "p=550,286;i=switch1_s";
        ap4: AccessPoint;
            gatesizes:
                in[1],
                out[1];
            display: "p=700,286;i=switch1_s";
        ap5: AccessPoint;
            gatesizes:
                in[1],
                out[1];
            display: "p=850,286;i=switch1_s";
        ap6: AccessPoint;
            gatesizes:
                in[1],
                out[1];
            display: "p=1000,286;i=switch1_s";
        ap7: AccessPoint;
            gatesizes:
                in[1],
                out[1];
            display: "p=1150,286;i=switch1_s";
        ap8: AccessPoint;
            gatesizes:
                in[1],
                out[1];
            display: "p=1300,286;i=switch1_s";
        ap9: AccessPoint;
            gatesizes:
                in[1],
                out[1];
            display: "p=1450,286;i=switch1_s";
        ap10: AccessPoint;
            gatesizes:
                in[1],
                out[1];
            display: "p=1600,286;i=switch1_s";
        ha: Router6;
            gatesizes:
                in[2],
                out[2];
```

```
            display: "p=72,56;i=router";
        hap: AccessPoint;
            gatesizes:
                in[1],
                out[1];
            display: "p=100,286;i=switch1_s";
        router: Router6;
            gatesizes:
                in[3],
                out[3];
            display: "p=280,136;i=router";
    connections nocheck:
        ar.in[0] <-- MIPv6SimpleIntranetCable <-- ap1.out[0];
        ar.out[0] --> MIPv6SimpleIntranetCable --> ap1.in[0];

        ap2.out[0] --> MIPv6SimpleIntranetCable --> ar.in[1];
        ap2.in[0] <-- MIPv6SimpleIntranetCable <-- ar.out[1];

        ap3.out[0] --> MIPv6SimpleIntranetCable --> ar.in[2];
        ap3.in[0] <-- MIPv6SimpleIntranetCable <-- ar.out[2];

        ap4.out[0] --> MIPv6SimpleIntranetCable --> ar.in[3];
        ap4.in[0] <-- MIPv6SimpleIntranetCable <-- ar.out[3];

        ap5.out[0] --> MIPv6SimpleIntranetCable --> ar.in[4];
        ap5.in[0] <-- MIPv6SimpleIntranetCable <-- ar.out[4];

        ap6.out[0] --> MIPv6SimpleIntranetCable --> ar.in[5];
        ap6.in[0] <-- MIPv6SimpleIntranetCable <-- ar.out[5];

        ap7.out[0] --> MIPv6SimpleIntranetCable --> ar.in[6];
        ap7.in[0] <-- MIPv6SimpleIntranetCable <-- ar.out[6];

        ap8.out[0] --> MIPv6SimpleIntranetCable --> ar.in[7];
        ap8.in[0] <-- MIPv6SimpleIntranetCable <-- ar.out[7];

        ap9.out[0] --> MIPv6SimpleIntranetCable --> ar.in[8];
        ap9.in[0] <-- MIPv6SimpleIntranetCable <-- ar.out[8];

        ap10.out[0] --> MIPv6SimpleIntranetCable --> ar.in[9];
        ap10.in[0] <-- MIPv6SimpleIntranetCable <-- ar.out[9];

        router.out[0] --> MIPv6SimpleInternetCable --> server.in[0];
        router.in[0] <-- MIPv6SimpleInternetCable <-- server.out[0];

        ar.out[10] --> LMACable --> router.in[2];
        ar.in[10] <-- LMACable <-- router.out[2];

       router.out[1] --> MIPv6SimpleInternetCable --> ha.in[0];
       router.in[1] <-- MIPv6SimpleInternetCable <-- ha.out[0];

        hap.out[0] --> MIPv6SimpleIntranetCable --> ha.in[1];
        hap.in[0] <-- MIPv6SimpleIntranetCable <-- ha.out[1];

    display: "p=2,10;b=1650,411";
endmodule

network L3HO : L3HO
endnetwork
```

Το αρχείο αρχικοποιήσεων και λειτουργίας (L3HO.ini)

```
#% old-wildcards
[General]
preload-ned-files=*.ned @../../../nedfiles.lst
network = L3HO

total-stack-kb=7535
ini-warnings = no
warnings = yes
rng-class=cLCG32
seed-0-lcg32 = seed_NR

[Cmdenv]
default-run=1
module-messages = no =
event-banners=no

[Tkenv]
breakpoints-enabled = no
animation-speed = 2.0


[Run 1]
sim-time-limit = 1615
L3HO.client1.pingApp.startTime=20
L3HO.client1.pingApp.destAddr = "server[0]"
L3HO.client1.pingApp.interval = 0.01
L3HO.client1.pingApp.printPing = False
L3HO.client1.pingApp.stopTime=1610
L3HO.client1*.mobilityManager.mobilityHandler.moveXmlConfig=xmldoc("L3HO.xml",
"netconf/global/ObjectMovement/MovingNode[0]")

[Run 2]
sim-time-limit = 815
L3HO.client1.pingApp.startTime=20
L3HO.client1.pingApp.destAddr = "server[0]"
L3HO.client1.pingApp.interval = 0.01
L3HO.client1.pingApp.printPing = False
L3HO.client1.pingApp.stopTime=810
L3HO.client1*.mobilityManager.mobilityHandler.moveXmlConfig=xmldoc("L3HO.xml",
"netconf/global/ObjectMovement/MovingNode[1]")

[Run 3]
sim-time-limit = 550
L3HO.client1.pingApp.startTime=20
L3HO.client1.pingApp.destAddr = "server[0]"
L3HO.client1.pingApp.interval = 0.01
L3HO.client1.pingApp.printPing = False
L3HO.client1.pingApp.stopTime=545
L3HO.client1*.mobilityManager.mobilityHandler.moveXmlConfig=xmldoc("L3HO.xml",
"netconf/global/ObjectMovement/MovingNode[2]")

[Run 4]
sim-time-limit = 415
L3HO.client1.pingApp.startTime=20
L3HO.client1.pingApp.destAddr = "server[0]"
L3HO.client1.pingApp.interval = 0.01
L3HO.client1.pingApp.printPing = False
L3HO.client1.pingApp.stopTime=410
L3HO.client1*.mobilityManager.mobilityHandler.moveXmlConfig=xmldoc("L3HO.xml",
"netconf/global/ObjectMovement/MovingNode[3]")
```

```
[Run 5]
sim-time-limit = 335
L3HO.client1.pingApp.startTime=20
L3HO.client1.pingApp.destAddr = "server[0]"
L3HO.client1.pingApp.interval = 0.01
L3HO.client1.pingApp.printPing = False
L3HO.client1.pingApp.stopTime=330
L3HO.client1*.mobilityManager.mobilityHandler.moveXmlConfig=xmldoc("L3HO.xml",
"netconf/global/ObjectMovement/MovingNode[4]")

[Run 6]
sim-time-limit = 280
L3HO.client1.pingApp.startTime=20
L3HO.client1.pingApp.destAddr = "server[0]"
L3HO.client1.pingApp.interval = 0.01
L3HO.client1.pingApp.printPing = False
L3HO.client1.pingApp.stopTime=275
L3HO.client1*.mobilityManager.mobilityHandler.moveXmlConfig=xmldoc("L3HO.xml",
"netconf/global/ObjectMovement/MovingNode[5]")

[Run 7]
sim-time-limit = 245
L3HO.client1.pingApp.startTime= 20
L3HO.client1.pingApp.destAddr = "server[0]"
L3HO.client1.pingApp.interval = 0.01
L3HO.client1.pingApp.printPing = False
L3HO.client1.pingApp.stopTime=240
L3HO.client1*.mobilityManager.mobilityHandler.moveXmlConfig=xmldoc("L3HO.xml",
"netconf/global/ObjectMovement/MovingNode[6]")

[Run 8]
sim-time-limit = 215
L3HO.client1.pingApp.startTime=20
L3HO.client1.pingApp.destAddr = "server[0]"
L3HO.client1.pingApp.interval = 0.01
L3HO.client1.pingApp.printPing = False
L3HO.client1.pingApp.stopTime=210
L3HO.client1*.mobilityManager.mobilityHandler.moveXmlConfig=xmldoc("L3HO.xml",
"netconf/global/ObjectMovement/MovingNode[7]")

[Run 9]
sim-time-limit = 195
L3HO.client1.pingApp.startTime=20
L3HO.client1.pingApp.destAddr = "server[0]"
L3HO.client1.pingApp.interval = 0.01
L3HO.client1.pingApp.printPing = False
L3HO.client1.pingApp.stopTime=190
L3HO.client1*.mobilityManager.mobilityHandler.moveXmlConfig=xmldoc("L3HO.xml",
"netconf/global/ObjectMovement/MovingNode[8]")

[Run 10]
sim-time-limit = 175
L3HO.client1.pingApp.startTime=20
L3HO.client1.pingApp.destAddr = "server[0]"
L3HO.client1.pingApp.interval = 0.01
L3HO.client1.pingApp.printPing = False
L3HO.client1.pingApp.stopTime=170
L3HO.client1*.mobilityManager.mobilityHandler.moveXmlConfig=xmldoc("L3HO.xml",
"netconf/global/ObjectMovement/MovingNode[9]")

[Run 11]
sim-time-limit = 160
```

```
L3HO.client1.pingApp.startTime=20
L3HO.client1.pingApp.destAddr = "server[0]"
L3HO.client1.pingApp.interval = 0.01
L3HO.client1.pingApp.printPing = False
L3HO.client1.pingApp.stopTime=155
L3HO.client1*.mobilityManager.mobilityHandler.moveXmlConfig=xmldoc("L3HO.xml",
"netconf/global/ObjectMovement/MovingNode[10]")

[Run 12]
sim-time-limit = 150
L3HO.client1.pingApp.startTime=20
L3HO.client1.pingApp.destAddr = "server[0]"
L3HO.client1.pingApp.interval = 0.01
L3HO.client1.pingApp.printPing = False
L3HO.client1.pingApp.stopTime=145
L3HO.client1*.mobilityManager.mobilityHandler.moveXmlConfig=xmldoc("L3HO.xml",
"netconf/global/ObjectMovement/MovingNode[11]")

[Run 13]
sim-time-limit = 138
L3HO.client1.pingApp.startTime=20
L3HO.client1.pingApp.destAddr = "server[0]"
L3HO.client1.pingApp.interval = 0.01
L3HO.client1.pingApp.printPing = False
L3HO.client1.pingApp.stopTime=133
L3HO.client1*.mobilityManager.mobilityHandler.moveXmlConfig=xmldoc("L3HO.xml",
"netconf/global/ObjectMovement/MovingNode[12]")

[Run 14]
sim-time-limit = 130
L3HO.client1.pingApp.startTime=20
L3HO.client1.pingApp.destAddr = "server[0]"
L3HO.client1.pingApp.interval = 0.01
L3HO.client1.pingApp.printPing = False
L3HO.client1.pingApp.stopTime=124
L3HO.client1*.mobilityManager.mobilityHandler.moveXmlConfig=xmldoc("L3HO.xml",
"netconf/global/ObjectMovement/MovingNode[13]")

[Run 15]
sim-time-limit = 120
L3HO.client1.pingApp.startTime=20
L3HO.client1.pingApp.destAddr = "server[0]"
L3HO.client1.pingApp.interval = 0.01
L3HO.client1.pingApp.printPing = False
L3HO.client1.pingApp.stopTime=115
L3HO.client1*.mobilityManager.mobilityHandler.moveXmlConfig=xmldoc("L3HO.xml",
"netconf/global/ObjectMovement/MovingNode[14]")

[Run 16]
sim-time-limit = 115
L3HO.client1.pingApp.startTime=20
L3HO.client1.pingApp.destAddr = "server[0]"
L3HO.client1.pingApp.interval = 0.01
L3HO.client1.pingApp.printPing = False
L3HO.client1.pingApp.stopTime=110
L3HO.client1*.mobilityManager.mobilityHandler.moveXmlConfig=xmldoc("L3HO.xml",
"netconf/global/ObjectMovement/MovingNode[15]")

[Run 17]
sim-time-limit = 110
L3HO.client1.pingApp.startTime=20
L3HO.client1.pingApp.destAddr = "server[0]"
L3HO.client1.pingApp.interval = 0.01
```

```
L3HO.client1.pingApp.printPing = False
L3HO.client1.pingApp.stopTime=105
L3HO.client1*.mobilityManager.mobilityHandler.moveXmlConfig=xmldoc("L3HO.xml",
"netconf/global/ObjectMovement/MovingNode[16]")

[Run 18]
sim-time-limit = 105
L3HO.client1.pingApp.startTime=20
L3HO.client1.pingApp.destAddr = "server[0]"
L3HO.client1.pingApp.interval = 0.01
L3HO.client1.pingApp.printPing = False
L3HO.client1.pingApp.stopTime=100
L3HO.client1*.mobilityManager.mobilityHandler.moveXmlConfig=xmldoc("L3HO.xml",
"netconf/global/ObjectMovement/MovingNode[17]")

[Run 19]
sim-time-limit = 100
L3HO.client1.pingApp.startTime=20
L3HO.client1.pingApp.destAddr = "server[0]"
L3HO.client1.pingApp.interval = 0.01
L3HO.client1.pingApp.printPing = False
L3HO.client1.pingApp.stopTime=95
L3HO.client1*.mobilityManager.mobilityHandler.moveXmlConfig=xmldoc("L3HO.xml",
"netconf/global/ObjectMovement/MovingNode[18]")

[Run 20]
sim-time-limit = 95
L3HO.client1.pingApp.startTime=20
L3HO.client1.pingApp.destAddr = "server[0]"
L3HO.client1.pingApp.interval = 0.01
L3HO.client1.pingApp.printPing = False
L3HO.client1.pingApp.stopTime=90
L3HO.client1*.mobilityManager.mobilityHandler.moveXmlConfig=xmldoc("L3HO.xml",
"netconf/global/ObjectMovement/MovingNode[19]")

[Parameters]

*.client1.networkLayer.proc.forwarding.routingInfoDisplay = true
L3HO.client1.linkLayers[*].NWIName="WirelessEtherModule"
L3HO.ap?.ds[*].NWIName="EtherModuleAP"

L3HO.server.networkLayer.proc.ICMP.icmpv6Core.icmpRecordRequests = false
L3HO.client1.networkLayer.proc.ICMP.icmpv6Core.icmpRecordRequests = false
L3HO.server.networkLayer.proc.ICMP.icmpv6Core.replyToICMPRequests = true

*.ha.linkLayers[0].NWIName="IPv6PPPInterface"
*.ar.linkLayers[10].NWIName="IPv6PPPInterface"
*.router.linkLayers[*].NWIName="IPv6PPPInterface"
*.server.linkLayers[0].NWIName="IPv6PPPInterface"


L3HO.*.IPv6routingFile = xmldoc("L3HO.xml")
*.networkInterface.txPower = 1.5
*.ap?.networkInterface.beaconPeriod = 0.1
*.ap?.networkInterface.authWaitEntryTimeout = 2
*.ap?.networkInterface.authEntryTimeout = 2
*.ap?.networkInterface.assEntryTimeout = 120
**.networkInterface.linkUpTrigger = false
**.networkInterface.retry = 7

include ../../../Etc/default.ini
```

Το αρχείο παραμετροποίησης του μοντέλου μας (L3HO.xml)

```
<?xml version="1.0" encoding="iso-8859-1"?>
<!DOCTYPE netconf SYSTEM "../../../Etc/netconf2.dtd">

<netconf debugChannel="HMIPv6Simple.log:rcfile:MobileMove:notice">
<!--
Ping6:Statistic:HMIPv6:custom:MIPv6MissedAdv:HMIPv6:AddrResln:MIPv6:AddressTimer:Route
rDisc:Forwarding:NeighbourDisc:debug">-->
  <global>
    <ObjectMovement>
     <MovingNode NodeName="pingpong" startTime="0">
        <move moveToX="1650" moveToY="300" moveSpeed="1"/>
      </MovingNode>
     <MovingNode NodeName="pingpong" startTime="0">
        <move moveToX="1650" moveToY="300" moveSpeed="2"/>
      </MovingNode>
     <MovingNode NodeName="pingpong" startTime="0">
        <move moveToX="1650" moveToY="300" moveSpeed="3"/>
      </MovingNode>
     <MovingNode NodeName="pingpong" startTime="0">
        <move moveToX="1650" moveToY="300" moveSpeed="4"/>
      </MovingNode>
     <MovingNode NodeName="pingpong" startTime="0">
        <move moveToX="1650" moveToY="300" moveSpeed="5"/>
      </MovingNode>
     <MovingNode NodeName="pingpong" startTime="0">
        <move moveToX="1650" moveToY="300" moveSpeed="6"/>
      </MovingNode>
     <MovingNode NodeName="pingpong" startTime="0">
        <move moveToX="1650" moveToY="300" moveSpeed="7"/>
      </MovingNode>
     <MovingNode NodeName="pingpong" startTime="0">
        <move moveToX="1650" moveToY="300" moveSpeed="8"/>
      </MovingNode>
     <MovingNode NodeName="pingpong" startTime="0">
        <move moveToX="1650" moveToY="300" moveSpeed="9"/>
      </MovingNode>
     <MovingNode NodeName="pingpong" startTime="0">
        <move moveToX="1650" moveToY="300" moveSpeed="10"/>
      </MovingNode>
     <MovingNode NodeName="pingpong" startTime="0">
        <move moveToX="1650" moveToY="300" moveSpeed="11"/>
      </MovingNode>
     <MovingNode NodeName="pingpong" startTime="0">
        <move moveToX="1650" moveToY="300" moveSpeed="12"/>
      </MovingNode>
     <MovingNode NodeName="pingpong" startTime="0">
        <move moveToX="1650" moveToY="300" moveSpeed="13"/>
      </MovingNode>
     <MovingNode NodeName="pingpong" startTime="0">
        <move moveToX="1650" moveToY="300" moveSpeed="14"/>
      </MovingNode>
     <MovingNode NodeName="pingpong" startTime="0">
        <move moveToX="1650" moveToY="300" moveSpeed="15"/>
      </MovingNode>
     <MovingNode NodeName="pingpong" startTime="0">
        <move moveToX="1650" moveToY="300" moveSpeed="16"/>
      </MovingNode>
     <MovingNode NodeName="pingpong" startTime="0">
        <move moveToX="1650" moveToY="300" moveSpeed="17"/>
```

```
      </MovingNode>
     <MovingNode NodeName="pingpong" startTime="0">
        <move moveToX="1650" moveToY="300" moveSpeed="18"/>
      </MovingNode>
     <MovingNode NodeName="pingpong" startTime="0">
        <move moveToX="1650" moveToY="300" moveSpeed="19"/>
      </MovingNode>
     <MovingNode NodeName="pingpong" startTime="0">
        <move moveToX="1650" moveToY="300" moveSpeed="20"/>
      </MovingNode>
    </ObjectMovement>
  </global>

 <local node="client1" mobileIPv6Support="on" mobileIPv6Role="MobileNode"
hierarchicalMIPv6Support="off" routeOptimisation="on" optimisticDAD="off">
<!-- HostDupAddrD... does not get read into InterfaceEntry properly in fact -->
<!-- perhaps other values do not work either (applies to xerces-c interface) -->
    <interface name="wlan0" HostDupAddrDetectTransmits="1">
    </interface>
  </local>
  <local node="server" mobileIPv6Support="on">
    <interface name="ppp0">
      <inetAddr>3011:bbbb:3333:6666:ac24:aff:fe11:bba</inetAddr>
    </interface>
  </local>

<!-- routing table configuration for primary HA -->
  <local node="ha" routePackets="on" mobileIPv6Support="on"
mobileIPv6Role="HomeAgent">
    <interface name="ppp0">
      <inetAddr>3018:EEEE:0:0:89d6:9cff:fe7e:83d2</inetAddr>
    </interface>
    <interface name="eth1" AdvSendAdvertisements="on" AdvHomeAgent="on">
      <AdvPrefixList>
       <AdvPrefix AdvOnLinkFlag="on"
AdvRtrAddrFlag="on">3018:EEEE:0:0:89d6:9cff:fe7e:83d2/64</AdvPrefix>
      </AdvPrefixList>
    </interface>

    <route>
      <routeEntry routeIface="ppp0" routeDestination="0/0"
       routeNextHop="3018:AAAA:0:1:4609:52ff:fe8b:a252"/>
      <routeEntry routeIface="eth1" routeDestination="3018:EEEE:0:0:0:0:0:0/64"/>
    </route>
  </local>
<!-- routing table configuration for AR
  Note: Does not require any hmip or mip support to forward map options. By default
all routers will forward received map options on all ifaces that are advertising.
  -->
  <local node="ar" routePackets="on" mobileIPv6Support="on"
mobileIPv6Role="HomeAgent">
    <interface name="eth0" AdvSendAdvertisements="on" AdvHomeAgent="on">
      <inetAddr>3018:FFFF:0:0:127b:c0ff:fe2e:7212</inetAddr>
      <AdvPrefixList>
       <AdvPrefix AdvOnLinkFlag="on"
AdvRtrAddrFlag="on">3018:FFFF:0:0:127b:c0ff:fe2e:7212/64</AdvPrefix>
      </AdvPrefixList>
    </interface>
    <interface name="eth1" AdvSendAdvertisements="on" AdvHomeAgent="on">
      <inetAddr>3018:FFFF:0:1:606:98ff:fe24:52f5</inetAddr>
      <AdvPrefixList>
       <AdvPrefix AdvOnLinkFlag="on"
AdvRtrAddrFlag="on">3018:FFFF:0:1:606:98ff:fe24:52f5/64</AdvPrefix>
```

```
      </AdvPrefixList>
    </interface>
    <interface name="eth2" AdvSendAdvertisements="on" AdvHomeAgent="on">
      <inetAddr>3018:FFFF:0:2:8087:eff:fe1a:7281</inetAddr>
      <AdvPrefixList>
        <AdvPrefix AdvOnLinkFlag="on"
AdvRtrAddrFlag="on">3018:FFFF:0:2:8087:eff:fe1a:7281/64</AdvPrefix>
       </AdvPrefixList>
    </interface>
    <interface name="eth3" AdvSendAdvertisements="on" AdvHomeAgent="on">
      <inetAddr>3018:FFFF:0:3:5f6a:a9ff:fe2c:df2e</inetAddr>
      <AdvPrefixList>
        <AdvPrefix AdvOnLinkFlag="on"
AdvRtrAddrFlag="on">3018:FFFF:0:3:5f6a:a9ff:fe2c:df2e/64</AdvPrefix>
       </AdvPrefixList>
    </interface>
    <interface name="eth4" AdvSendAdvertisements="on" AdvHomeAgent="on">
      <inetAddr>3018:FFFF:0:4:2145:bc34:fe4b:df2f</inetAddr>
      <AdvPrefixList>
        <AdvPrefix AdvOnLinkFlag="on"
AdvRtrAddrFlag="on">3018:FFFF:0:4:2145:bc34:fe4b:df2f/64</AdvPrefix>
       </AdvPrefixList>
    </interface>
    <interface name="eth5" AdvSendAdvertisements="on" AdvHomeAgent="on">
      <inetAddr>3018:FFFF:0:5:5aca:a9f:fe4f:d3ae</inetAddr>
      <AdvPrefixList>
        <AdvPrefix AdvOnLinkFlag="on"
AdvRtrAddrFlag="on">3018:FFFF:0:5:5aca:a9f:fe4f:d3ae/64</AdvPrefix>
       </AdvPrefixList>
    </interface>
    <interface name="eth6" AdvSendAdvertisements="on" AdvHomeAgent="on">
      <inetAddr>3018:FFFF:0:6:215a:a34f:feaa:daae</inetAddr>
      <AdvPrefixList>
        <AdvPrefix AdvOnLinkFlag="on"
AdvRtrAddrFlag="on">3018:FFFF:0:6:215a:a34f:feaa:daae/64</AdvPrefix>
       </AdvPrefixList>
    </interface>
    <interface name="eth7" AdvSendAdvertisements="on" AdvHomeAgent="on">
      <inetAddr>3018:FFFF:0:7:25a:a32f:faaa:d23e</inetAddr>
      <AdvPrefixList>
        <AdvPrefix AdvOnLinkFlag="on"
AdvRtrAddrFlag="on">3018:FFFF:0:7:25a:a32f:faaa:d23e/64</AdvPrefix>
       </AdvPrefixList>
    </interface>
    <interface name="eth8" AdvSendAdvertisements="on" AdvHomeAgent="on">
      <inetAddr>3018:FFFF:0:8:244a:dc4f:fe12:1aae</inetAddr>
      <AdvPrefixList>
        <AdvPrefix AdvOnLinkFlag="on"
AdvRtrAddrFlag="on">3018:FFFF:0:8:244a:dc4f:fe12:1aae/64</AdvPrefix>
       </AdvPrefixList>
    </interface>
    <interface name="eth9" AdvSendAdvertisements="on" AdvHomeAgent="on">
      <inetAddr>3018:FFFF:0:9:21ff:6dcf:87a:da6e</inetAddr>
      <AdvPrefixList>
        <AdvPrefix AdvOnLinkFlag="on"
AdvRtrAddrFlag="on">3018:FFFF:0:9:21ff:6dcf:87a:da6e/64</AdvPrefix>
       </AdvPrefixList>
    </interface>
    <interface name="ppp10">
      <!-- does not need to be globally scoped -->
      <inetAddr>3018:FFFF:0:4:5f6a:a9ff:fe2c:df2f</inetAddr>
    </interface>
    <route>
```

```
      <routeEntry
       routeIface="eth0"
       routeDestination="3018:FFFF:0:0:0:0:0:0/64"/>
      <routeEntry
       routeIface="eth1"
       routeDestination="3018:FFFF:0:1:0:0:0:0/64"/>
      <routeEntry
       routeIface="eth2"
       routeDestination="3018:FFFF:0:2:0:0:0:0/64"/>
      <routeEntry
       routeIface="eth3"
       routeDestination="3018:FFFF:0:3:0:0:0:0/64"/>
      <routeEntry
       routeIface="eth4"
       routeDestination="3018:FFFF:0:4:0:0:0:0/64"/>
      <routeEntry
       routeIface="eth5"
       routeDestination="3018:FFFF:0:5:0:0:0:0/64"/>
      <routeEntry
       routeIface="eth6"
       routeDestination="3018:FFFF:0:6:0:0:0:0/64"/>
      <routeEntry
       routeIface="eth7"
       routeDestination="3018:FFFF:0:7:0:0:0:0/64"/>
      <routeEntry
       routeIface="eth8"
       routeDestination="3018:FFFF:0:8:0:0:0:0/64"/>
      <routeEntry
       routeIface="eth9"
       routeDestination="3018:FFFF:0:9:0:0:0:0/64"/>
      <routeEntry
       routeIface="ppp10" routeNextHop="3018:AAAA:0:2:0450:90ff:fe5d:f971"
       routeDestination="0/0"/>

    </route>
  </local>
<!-- routing table configuration for MAP -->
 <local node="router" routePackets="on" mobileIPv6Support="on"
mobileIPv6Role="HomeAgent" hierarchicalMIPv6Support="off">
    <interface name="ppp0"  AdvSendAdvertisements="on">
      <AdvPrefixList>
        <AdvPrefix AdvOnLinkFlag="on">3011:BBBB:3333:6666:0:0:0:0/64</AdvPrefix>
      </AdvPrefixList>
    </interface>
   <interface name="ppp1">
 <!-- does not need to be globally scoped but needs to be assigned for static
      routing purposes -->
      <inetAddr>3018:AAAA:0:1:4609:52ff:fe8b:a252</inetAddr>
    </interface>
    <interface name="ppp2" AdvSendAdvertisements="on" AdvHomeAgent="on">
      <inetAddr>3018:AAAA:0:2:0450:90ff:fe5d:f971</inetAddr>
      <AdvPrefixList>
        <AdvPrefix>3018:AAAA:0:2:0450:90ff:fe5d:f971</AdvPrefix>
      </AdvPrefixList>
    </interface>
    <route>
      <!-- to server -->
      <routeEntry
       routeIface="ppp0" routeDestination="3011:BBBB:3333:6666:0:0:0:0/64"/>
      <!-- Goes to primary HA -->
      <routeEntry
       routeIface="ppp1" routeDestination="3018:EEEE:0:0:0:0:0:0/32"
       routeNextHop="3018:EEEE:0:0:89d6:9cff:fe7e:83d2"/>
```

```
      <!-- Goes to 4th if of AR -->
      <routeEntry routeIface="ppp2" routeDestination="0/0"
       routeNextHop="3018:FFFF:0:4:5f6a:a9ff:fe2c:df2f"/>
    </route>
  </local>

</netconf>
```

Το αρχείο παραμετροποίησης αναφέρεται στην βασική MIPv6 τεχνική. Η αλλαγή σε κάποια άλλη τεχνική γίνεται αλλάζοντας ή εισάγοντας κάποιες επιπλέον παραμέτρους στο XML αρχείο.

Πιο συγκεκριμένα:

- **FSRA:** Σε κάθε interface των δρομολογητών προσθέτουμε την παράμετρο <MaxFastRAS="10">
- **ODAD:** Στις παραμέτρους του κινητού κόμβου προσθέτουμε την παράμετρο <optimisticDAD="on">
- **Fbeac:** Σε κάθε interface των δρομολογητών προσθέτουμε τα <MIPv6MaxRtrAdvInterval="0.07" MIPv6MinRtrAdvInterval="0.03">
- **EBU:** Στις παραμέτρους του κινητού κόμβου προσθέτουμε τα <earlyBU="on" returnRoutability="on">
- **HMIPv6:** Στις παραμέτρους του κινητού κόμβου προσθέτουμε το <hierarchicalMIPv6Support="on"> και στις παραμέτρους του δρομολογητή που ενεργεί ως MAP τα <hierarchicalMIPv6Support="on" map="on">
- **L2 Trigger:** Για την L2 Trigger δεν αλλάζουμε κάτι στο XML αρχείο, αλλά στο INI αρχείο, συγκεκριμένα <**.networkInterface.linkUpTrigger = true>

Συνδυάζοντας τα παραπάνω μπορούμε να ενεργοποιήσουμε οποιονδήποτε συνδιασμό.

Βιβλιογραφια και Πηγες

[1] Internet World Stats, Ιούλιος 2007

[2] Eric Zerman, General Internet Use Up, But What About Mobile Internet Usage?, Μάρτιος 2007

[3] C. Perkins, "RFC 3220 IP Mobility Support for IPv4", Ιανουάριος 2002

[4] J. Postel, "RFC 791 Internet Protocol", Σεπτέμβριος 1981

[5] C. Perkins, "RFC 2002 IP Mobility Support", Οκτώβριος 1996

[6] Wikipedia

[7] D Ghosh, Mobile IP

[8] C. Perkins, "RFC 2003 IP Encapsulation within IP", Οκτώβριος 1996

[9] R. Zipagan, R. Kheraj, "Concerns with Mobile IP"

[10] C. Perkins, D. B. Johnson, "Route Optimization in Mobile IP", Internet Draft, Νοέμβριος 1997

[11] S. Deering, R. Hinden "RFC 2460 Internet Protocol, Version 6 (IPv6) Specification", Δεκέμβριος 1998

[12] A.Conta, S.Deering, "RFC 2473 Generic tunneling in IPv6 specification", Δεκέμβριος 1998

[13] T. Narten, E. Nordmark, W. Simpson, "RFC 2461 Neighbor Discovery for IP Version 6 (IPv6)", Δεκέμβριος 1998

[14] R. Hinden, S. Deering, "RFC 2373 IP Version 6 Addressing Architecture", Ιούλιος 1998

[15] A. Conta, "RFC 3122 Extensions to IPv6 Neighbor Discovery for Inverse Discovery", Ιούνιος 2001

[16] S. Thomson, T. Narten, "RFC 2462 IPv6 Stateless Address Autoconfiguration", Δεκέμβριος 1998

[17] D. Johnson, C. Perkins, J. Arkko "RFC 3775 Mobility Support in IPv6", Ιούνιος 2004

[18] P. Ferguson, D. Senie, "RFC 2827 Network Ingress Filtering: Defeating Denial of Service Attacks which employ IP Source Address Spoofing", Μάιος 2000.

[19] J. Arkko, V. Devarapalli, F. Dupont, "RFC 3776 Using IPsec to Protect Mobile IPv6 Signaling Between Mobile Nodes and Home Agents", Ιούνιος 2004

[20] V. Devarapalli, F. Dupont , "RFC 4877 Mobile IPv6 Operation with IKEv2 and the Revised IPsec Architecture", Απρίλιος 2007

[21] R. Koodli, "RFC 4068 Fast Handovers for Mobile IPv6", Ιούλιος 2005

[22] J. Kempf, M. M. Khalil, "IPv6 Fast Router Advertisement", Internet Draft, Ιούλιος 2004

[23] N. Moore, "RFC 4429 Optimistic Duplicate Address Detection (DAD) for IPv6", Απρίλιος 2006

[24] D. B. Johnson, C. E. Perkins, J. Arkko, "Mobility Support in IPv6", Internet draft, Ιούνιος 2002

[25] J. Kempf, "RFC 4830 Problem Statement for Network-Based Localized Mobility Management (NETLMM)", Απρίλιος 2007

[26] J. Kempf, "RFC 4831 Goals for Network-Based Localized Mobility Management (NETLMM)", Απρίλιος 2007

[27] P. Eardley, N. Georganopoulos, M. West, "On the scalability of IP micro-mobility management protocols", in Proceedings of the 4th International Workshop on Mobile and Wireless Communications Network, 2002, pp. 470-474

[28] H. Soliman, C. Castelluccia, K. El Malki, L. Bellier, "RFC 4140 Hierarchical Mobile IPv6 Mobility Management (HMIPv6)", Αύγουστος 2005

[29] K. Kawano, K. Kinoshita, K. Murakami, "Multilevel Hierarchical Distributed IP Mobility Management Scheme for Wide Area Networks", in Proceedings of IEEE ICCCN, 2002

[30] E. Natalizio, A. Scicchitano, S. Marano, "Mobility Anchor Point Selection Based on User Mobility in HMIPv6 Integrated with Fast Handover Mechanism", in Preceedings of IEEE WCNC, 2005

[31] V. Thing, H. Lee, Y. Xu, "A Local Mobility Agent Selection Algorithm for Mobile Networks", IEEE International Conference on Communications, 2003

[32] S. Pack, M. Nam, T. Kwon, Y. Choi, "An adaptive mobility anchor point selection scheme in hierarchical Mobile IPv6 networks", Computer Communications, v29 i16 3066-3078, 2005

[33] H. Jung, H. Soliman, S. Koh, "Fast Handover for Hierarchical MIPv6 (F-HMIPv6)", Internet Draft, Οκτώβριος 2005

[34] H. Jung, S. Koh, "Fast handover support in hierarchical mobile IPv6", 6th International Conference on Advanced Communication Technology, 2004, Volume 2, Issue , 2004 pp 551-554

[35] H. Jung, E. Kim, J. YI, H. Lee, "A Scheme for Supporting Fast Handover in Hierarchical Mobile IPv6 Networks", ETRI Journal, vol.27, no.6, Δεκέμβριος 2005, pp.798-801.

[36] A. Campbell, J. Gomez, C. Wan, S. Kim, Z. Turanyi, A. Valko, "Cellular IP", Internet Draft, Ιούλιος 2000

[37] R. Ramjee, T. La Porta, S. Thuel, K. Varadhan, L. Salgarelli, "IP micro-mobility support using HAWAII", Internet Draft, Ιανουάριος 2000

[38] A. Misra, S. Das, A. Mcauley, A. Dutta, S. K. Das, "IDMP: An Intra-Domain Mobility Management Protocol using Mobility Agents", Ιούλιος 2000

[40] A. O'Neill, G. Tsirtsis, S. Corson, "Edge Mobility Architecture", Internet Draft, Ιούλιος 2000

[41] V. Park, S. Corson, "Temporally-Ordered Routing Algorithm (TORA)", Internet Draft, Ιούλιος 2001

[42] OMNeT++ Discrete Event Simulation System

[43] A. Varga, "OMNeT++ 3.2 User Manual"

[44] J. Lai, E. Wu, IPv6Suite Simulation Framework

[45] INET Framework

[46] R. Hinden, M. O'Dell, S. Deering, "RFC 2374 An IPv6 Aggregatable Global Unicast Address Format", Ιούλιος 1998

[47] A. Conta, S. Deering, M. Gupta, "RFC 4443 Internet Control Message Protocol (ICMPv6) for the Internet Protocol Version 6 (IPv6)", Μάρτιος 2006

[48] M. Crawford, "RFC 2464 Transmission of IPv6 Packets over Ethernet Networks", Δεκέμβριος 1998

[49] D. Haskin, E. Allen, "RFC 2472 IP Version 6 over PPP", Δεκέμβριος 1998

[50] H. Soliman, C. Catelluccia, K. El Malki, L. Bellier, "Hierarchical Mobile IPv6 mobility management (HMIPv6)", Internet Draft, Ιούνιος 2004

[51] N. Moore, "Optimistic Duplicate Address Detection", Internet Draft, Μάρτιος 2004

[52] J. Kempf, M. Khalil, "IPv6 Fast Router Advertisement", Internet Draft, Οκτώβριος 2004

[53] C. Vogt, "Early Binding Updates for Mobile IPv6", Internet Draft, Φεβρουάριος 2005

[54] T. Narten, E. Nordmark, W. Simpson, H. Soliman, J. Tatuya, "Neighbor Discovery for IP version 6 (IPv6)", Internet Draft, Οκτώβριος 2003

[55] J. Lai, "Performance Evaluation of Mobility Management Protocols for the Next Generation Internet (IPv6)", Master Thesis, Monash University, Ιανουάριος 2004

[56] C. Vogt, R. Bless, M. Doll, T. K'fner, "Early Binding Updates for Mobile IPv6", Internet draft, Φεβρουάριος 2004