ΤΕΧΝΟΛΟΓΙΚΟ ΕΚΠΑΙΔΕΥΤΙΚΟ ΙΔΡΥΜΑ ΚΡΗΤΗΣ
ΤΜΗΜΑ ΗΛΕΚΤΡΟΝΙΚΗΣ

ΔΙΑΧΕΙΡΙΣΗ ΚΙΝΗΤΙΚΟΤΗΤΑΣ ΣΤΑ ΑΣΥΡΜΑΤΑ ΔΙΚΤΥΑ

ΜΑΛΑΝΔΡΑΚΗΣ Χ. ΓΕΩΡΓΙΟΣ

ΕΙΣΗΓΗΤΗΣ
ΛΙΟΔΑΚΗΣ Σ. ΓΕΩΡΓΙΟΣ
Καθηγητής Εφαρμογών

Abstract

The basic mobility management is composed of two types, location management and hancover management. The location management concerns the discovery of the current point of attachment of a mobile user for the delivery of incoming calls. The handover management is concerned with maintining connections of the mobile host continue to more and change its point of attachment to the network.

The emphasis of this thesis is a handover management and its constituent phases, i.e. handover initiation and handover execution. In particular, various approaches examined by the research community are reviewed as well as the framework envisaged by the Mobile IP protocol. Moreover, the exploitation of link layer triggering (L2 triggering) for handover performance optimization is presented and studied through simulation.

The simulation model was implemented in the Omnet++ environment by the use of the open source IPv6Suite. Performance results, as indicated by the handover latency and the associated packet loss, show that MobileIPv6 mechanisms for handover initiation should be enhanced by L2 triggering.

Περιεχόμενα

1. Εισαγωγή

Την τελευταία δεκαετία, γίνεται όλο και περισσότερο επιτακτική η ανάγκη για ασύρματη πρόσβαση στο Διαδίκτυο. Σε αυτό βέβαια συντελεί και το γεγονός ότι φορητά υπολογιστικά συστήματα όπως φορητοί υπολογιστές, υπολογιστές παλάμης και κινητά τηλέφωνα έχουν φθάσει σε προσιτές τιμές για τον μέσο χρήστη.

Σημαντικός παράγοντας όμως είναι και η ίδια η φύση του Internet. Στα πρώτα βήματά του, η τοπολογία του ήταν «κεντροστρεφής» , δηλαδή λίγοι πανίσχυροι κεντρικοί κόμβοι με τους οποίους θα συνδέονταν πλήθος απλών τερματικών. Οι κεντρικοί κόμβοι θα αναλάμβαναν την αποθήκευση και διαχείριση των δεδομένων και τα τερματικά θα περιορίζονταν μόνο στην επικοινωνία με τους κεντρικούς κόμβους. Το παραπάνω σενάριο μπορεί να είναι ρεαλιστικό για μικρά δίκτυα, αλλά σαφώς δεν αντιπροσωπεύει την σημερινή δομή του Internet.

Σήμερα, η δομή του Internet βασίζεται σε μια κατανεμημένη τοπολογία, όπου οι τελικοί σταθμοί–κόμβοι έχουν αρκετή υπολογιστική ισχύ και δυνατότητες, ώστε να μην εξαρτώνται από κεντρικούς υπολογιστές. Το Διαδίκτυο δηλαδή είναι παντού και δεν απαρτίζεται μόνο από κλασικούς υπολογιστές αλλά και από κινητά τηλέφωνα, φορητούς υπολογιστές και άλλες συσκευές.

Η σύνδεση όμως των παραπάνω συσκευών στο Διαδίκτυο, σύμφωνα με τον κλασικό τρόπο λειτουργίας του, δεν είναι εφικτή σε τέτοιο βαθμό ώστε να μιλάμε για *επικοινωνία οπουδήποτε, οποτεδήποτε με οποιονδήποτε (communicating anywhere, anytime with anyone).* Η αδυναμία της οικογένειας πρωτοκόλλων TCP/IP να υποστηρίξουν αυτό το οποίο λέγεται **mobile computing** οδήγησε στην ανάγκη ανάπτυξης άλλων πρωτοκόλλων, τα οποία είναι στην ουσία επεκτάσεις του Internet Protocol (IP). Σκοπός των πρωτοκόλλων αυτών είναι να επιτευχθεί η **κινητικότητα** υπολογιστικών συσκευών στο Διαδίκτυο.

Οι προτάσεις που έχουν γίνει για την υποστήριξη της κινητικότητας υπολογιστικών συστημάτων παρουσιάζονται και αναλύονται στην παρούσα εργασία και δίδεται ιδιαίτερη έμφαση στην διαδικασία του **handover** όπου αποτελεί και ένα

από τα σημαντικότερα ζητήματα στη διαχείριση κινητικότητας στα ασύρματα δίκτυα.

1.1 Κινητικότητα (Mobility)

Με τον όρο κινητικότητα, εννοούμε την δυνατότητα ενός κόμβου να αλλάζει το σημείο πρόσβασής του στο δίκτυο χωρίς να διακόπτονται οι υπάρχουσες συνδέσεις. Αξίζει να σημειωθεί ότι όσον αφορά τα ασύρματα IP δίκτυα, τα θέματα της Διαχείρισης Κινητικότητας (Mobility Management) σε συνδυασμό με την Ποιότητα Υπηρεσιών (Quality of Service) αποτελούν τα ζητήματα έρευνας και ανάπτυξης σήμερα.

Για να μπορούμε όμως να κατανοήσουμε καλύτερα την Διαχείριση Κινητικότητας, πρέπει να μελετήσουμε τα διάφορα σενάρια ασύρματης πρόσβασης.

1.2 Κινητικότητα και Μεταφερσιμότητα

Εδώ πρέπει να ξεκαθαρίσουμε την διαφορά ανάμεσα στις έννοιες mobility (κινητικότητα) και portability (μεταφερσιμότητα). Με τον όρο κινητικότητα αναφερόμαστε στην κίνηση ενός κινητού σταθμού αλλάζοντας σημεία πρόσβασης χωρίς όμως να διακόπτονται οι υπάρχουσες συνδέσεις.

Αντίθετα με τον όρο portability, εννοούμε την κίνηση σε διαφορετικά δίκτυα, όπου έχουμε όμως διακοπή και επανεκκίνηση των συνδέσεων του τελικού σταθμού.

Κατά την διάρκεια της έρευνας και ανάπτυξης που έγινε στο θέμα Διαχείριση Κινητικότητας, παρουσιάστηκε η αδυναμία αντιμετώπισης όλων των περιπτώσεων κινητικότητας με το ίδιο πρωτόκολλο. Έτσι αναδείχθηκαν δύο νέες έννοιες, μακρο-κινητικότητα (macromobility) και μικρο-κινητικότητα (micromobility). Με τον όρο macromobility εννοούμε την κίνηση του κινητού σταθμού από δίκτυα σε δίκτυα, ενώ με τον όρο micromobility ή μ-mobility εννοούμε την κίνηση του κινητού σταθμού ανάμεσα σε μικρά υποδίκτυα.

Το Mobile IP ήταν η πρώτη λύση που προτάθηκε με σκοπό να καλύψει όλες τις

περιπτώσεις κινητικότητας. Στη συνέχεια όμως, έγινε εμφανές ότι το Mobile IP δεν μπορεί να αντεπεξέλθει σε σενάρια μικροκινητικότητας (μ-mobility) για λόγους τους οποίους θα παρουσιάσουμε παρακάτω.

Οι νέες τάσεις που εφαρμόζονται στη διαχείριση κινητικότητας βασίζονται στην ιδέα ότι ο κύριο κορμός του δικτύου (backbone) θα βασίζεται στο Mobile IP (macro-mobility), ενώ στις περιπτώσεις micro-mobility θα γίνεται χρήση άλλων πρωτοκόλλων όπως Cellular IP, HAWAII και TIMIP τα οποία έχουν σχεδιαστεί ειδικά για αυτές τις περιπτώσεις.

1.3 Σενάρια Ασύρματης Πρόσβασης

Γενικά, τρία σενάρια για την χρήση ασύρματης πρόσβασης παρουσιάζονται:

Το πρώτο σενάριο, το οποίο αναφέρεται ως *βασική ασύρματη πρόσβαση (basic wireless access)* έχει να κάνει με την χρήση ασύρματων μέσων μετάδοσης κυρίως λόγω της αδυναμίας εγκατάστασης ενσύρματων μέσων. Κατά το σενάριο αυτό, οι τερματικοί σταθμοί βρίσκονται σε μία συγκεκριμένη τοποθεσία και συνδέονται με έναν μόνο και συνήθως τον ίδιο σταθμό βάσης, και κινούνται πολύ αργά έως και καθόλου.

Κλασικό παράδειγμα είναι η χρήση ασύρματων δικτύων στο γραφείο ή στο σπίτι. Η μεγαλύτερη πρόκληση σε αυτό το σενάριο, δεν είναι η Διαχείριση Κινητικότητας αλλά η Ποιότητα των Παρεχόμενων Υπηρεσιών (QoS). Αρκετές τεχνολογίες έχουν αναπτυχθεί για το συγκεκριμένο σενάριο ασύρματης μετάδοσης. Wireless LANs, Bluetooth και IrDA είναι λίγες αλλά χαρακτηριστικές από αυτές.

Κατά το δεύτερο σενάριο, το οποίο θα αναφέρουμε ως *νομαδική ασύρματη πρόσβαση (nomadic wireless access)* οι τερματικοί σταθμοί κινούνται σε αποστάσεις, κατά πολύ μεγαλύτερες από τα όρια ενός σημείου πρόσβασης. Οι σταθμοί βάσης, απέχουν αρκετά μεταξύ τους και δεν υπάρχει σε κανένα σημείο αλληλοεπικάλυψη σταθμών βάσεων.

Για να γίνει καλύτερα κατανοητό το συγκεκριμένο σενάριο, ας υποθέσουμε έναν χρήστη του οποίου, στον χώρο εργασίας του παρέχεται ασύρματη πρόσβαση στο

Διαδίκτυο. Ο χρήστης κάνει χρήση του ασύρματου δικτύου μέσω π.χ. του φορητού του υπολογιστή. Εγκαταλείποντας τον χώρο εργασίας του, δεν υπάρχει πλέον ασύρματη πρόσβαση. Στην συνέχεια θα μετακινηθεί σε κάποιον άλλο χώρο όπου παρέχεται πρόσβαση π.χ. σε κάποιο αεροδρόμιο και θα κάνει χρήση της εκεί ασύρματης πρόσβασης.

Όπως φαίνεται στο συγκεκριμένο σενάριο, δεν μπαίνουν θέματα handover και mobility management. Αντίθετα, άλλα θέματα που τίθενται προς συζήτηση είναι η δυνατότητα των κινητών σταθμών να αντεπεξέλθουν σε ετερογενής τεχνολογίες που πιθανών παρέχονται από τους σταθμούς βάσης, θέματα πιστοποίησης, χρέωσης κ.λ.π.

Τέλος, το τρίτο σενάριο, στο οποίο θα δίναμε την ονομασία *πραγματική κινητή πρόσβαση (true mobile access)*, έχουμε σταθμούς βάσης όπου οι περιοχές κάλυψής τους αλληλεπικαλύπτονται, και οι κινητοί σταθμοί εμφανίζουν μεγάλη κινητικότητα. Πιο συγκεκριμένα, οι κινητοί σταθμοί κατά την διάρκεια μιας επικοινωνίας αλλάζουν το σημείο πρόσβασής τους, οπότε και εμφανίζεται το handover.

Όπως καταλαβαίνουμε, η Διαχείριση Κινητικότητας αποτελεί σημαντικό ζήτημα για το εν λόγω σενάριο και ιδιαίτερα το θέμα του handover. Την αδιάλειπτη επικοινωνία του κινητού σταθμού με το Διαδίκτυο, εγγυάται ο σωστός αλγόριθμος handover. Στην παρούσα εργασία, θα ασχοληθούμε κυρίως με το εν λόγω σενάριο ασύρματης πρόσβασης, μια και αυτό είναι που απαιτεί αποδοτικούς αλγόριθμους handover και γενικότερα μια αποδοτική Διαχείριση Κινητικότητας.

1.4 Η δομή του Internet

Το κυρίως πρωτόκολλο του Διαδικτύου, είναι το Internet Protocol (IP).Το IP είναι ένα πρωτόκολλο μεταγωγής πακέτων (packet-switch protocol) το οποίο είναι υπεύθυνο για τη διευθυνσιοδότηση και την επιλογή δρομολογίου. Το IP βρίσκεται στη στρώση δικτύου του Μοντέλου Αναφοράς OSI (OSI Reference Model).

Στρώση Εφαρμογών
Στρώση Παρουσίασης
Στρώση Συνδιάλεξης
Στρώση Μεταφοράς
Στρώση Δικτύου
Στρώση Σύνδεσης Δεδομένων
Φυσική Στρώση

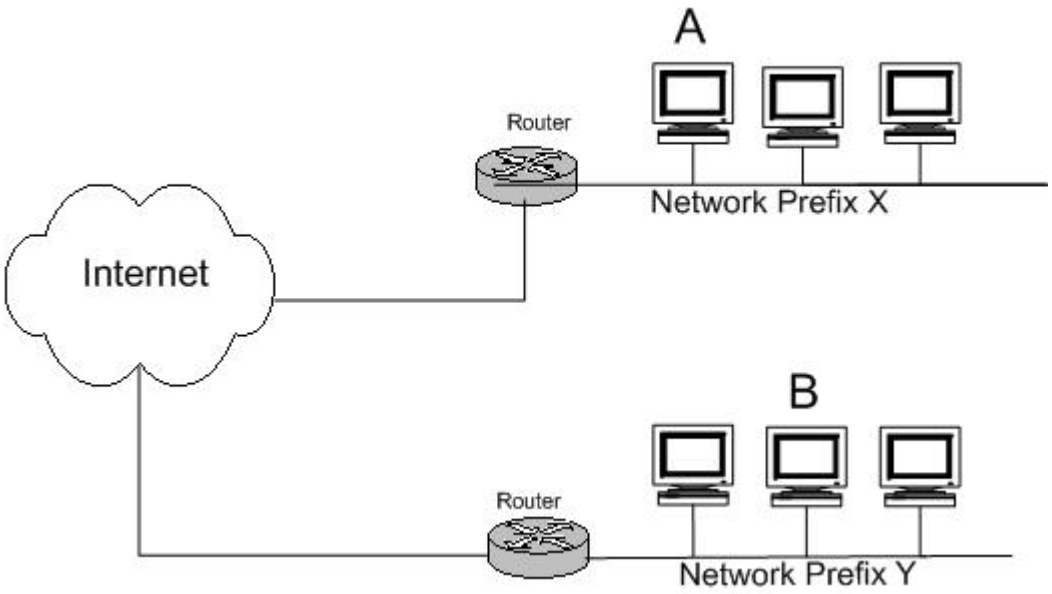
Σχήμα 1 – OSI Reference Model

Στο Διαδίκτυο, το πρωτόκολλο της στρώσης Μεταφοράς (Transport Layer), είναι είτε το «ασυνδεσμικό» UDP (User Datagram Protocol, Πρωτόκολλο Αυτοδύναμων Πακέτων Χρήστη) το οποίο δεν εγκαθιδρύει τη σύνδεση, απλώς προσπαθεί να στείλει τα δεδομένα και να επαληθεύσει ότι ο παραλήπτης – υπολογιστής πράγματι τα λαμβάνει, είτε το TCP (Transport Control Protocol, Πρωτόκολλο Ελέγχου Μετάδοσης) το οποίο είναι ένα πρωτόκολλο «προσανατολισμένο στη σύνδεση» και εγκαθιδρύει μια σύνδεση (session). Αξίζει να σημειωθεί ότι τα δύο αυτά πρωτόκολλα χρησιμοποιούν διαφορετικές θύρες (ports) οπότε είναι δυνατή η χρήση των δύο αυτών πρωτοκόλλων ταυτόχρονα χωρίς διενέξεις.

Όλοι οι κόμβοι (nodes) στο Internet, πρέπει να υποστηρίζουν το IP πρωτόκολλο. Ένας κόμβος αναγνωρίζεται από την IP address, που είναι ένας αριθμός που

αποτελείται από δύο μέρη: Το **network prefix** (πρόθεμα δικτύου) το οποίο ταυτοποιεί το δίκτυο στο οποίο βρίσκεται ο κόμβος, και το **host ID** το οποίο ταυτοποιεί τον κόμβο στο δίκτυο.

Το Internet αποτελείται από δεκάδες χιλιάδες δίκτυα, τα οποία συνδέονται μεταξύ τους από κόμβους με δυνατότητες δρομολόγησης, τους δρομολογητές (routers). Στο σχήμα 2, φαίνονται ο κόμβος A που βρίσκεται στο δίκτυο με network prefix X και ο κόμβος B ο οποίος βρίσκεται στο δίκτυο με network prefix Y.



Σχήμα 2

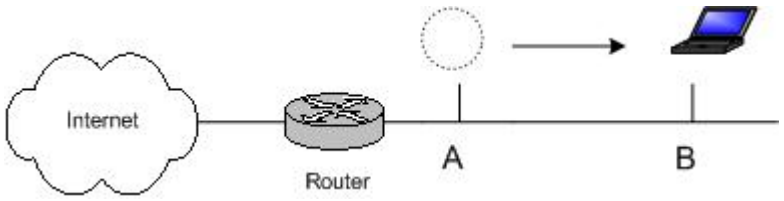
Στο παράδειγμά μας, ο κόμβος A στέλνει ένα πακέτο δεδομένων στον κόμβο B. Η διεύθυνση προορισμού, αποτελείται από το πρόθεμα δικτύου και το host ID. Η δρομολόγηση των πακέτων, από τους routers, γίνεται βάσει του network prefix. Το host ID δεν χρησιμοποιείται έως ότου το πακέτο φθάσει στο δίκτυο με πρόθεμα Y.

1.5 Σύνδεση κινητών σταθμών (mobile nodes) στο Διαδίκτυο

Με τον όρο κινητός σταθμός ή **κινητός κόμβος** (**mobile node**), εννοούμε τον κόμβο ο οποίος κινείται και παράλληλα είναι συνδεδεμένος με ένα δίκτυο. Παράδειγμα κινητού κόμβου, είναι ένας φορητός υπολογιστής ή ένα PDA. Ο mobile node κατά κανόνα, κάνει χρήση ασύρματων ζεύξεων για την πρόσβαση στο δίκτυο.

Ένας κόμβος είναι συνδεδεμένος σε ένα δίκτυο, μέσω ενός **σημείου πρόσβασης** (**access point**). Στην περίπτωση όπου το σημείο πρόσβασης παρέχει ασύρματη σύνδεση, τότε συνήθως λέγεται **σταθμός βάσης** (**base station**). Το σημείο πρόσβασης στο οποίο είναι συνδεδεμένος ο mobile node αναφέρεται και ως **point of attachment**.

Όταν ο mobile node κινείται και αλλάζει σημείο σύνδεσης, αλλά παραμένει στο ίδιο δίκτυο, είναι προφανές ότι η IP address παραμένει η ίδια και δεν χρειάζεται να αλλάξει ούτε το network prefix αλλά ούτε και το host ID. Το παραπάνω παράδειγμα φαίνεται στο σχήμα 3.

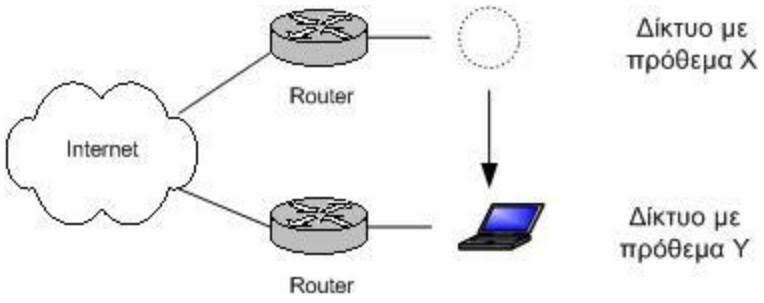


Σχήμα 3

Υπάρχει όμως και η περίπτωση, η κίνηση του mobile node να είναι από το point of attachment ενός δικτύου σε point of attachment άλλου δικτύου. Η υπόθεση αυτή φαίνεται στο σχήμα 4.

Στο σχήμα 4, ο mobile node αλλάζει το point of attachment του δικτύου με πρόθεμα X, σε δίκτυο με πρόθεμα Y. Στο συγκεκριμένο σενάριο, γίνονται αντιληπτές οι αδυναμίες του IP πρωτοκόλλου όσον αφορά την κινητικότητα.

Πιο συγκεκριμένα:



Σχήμα 4

1. Εάν ο mobile node, κρατήσει την IP address με network prefix X, τότε τα πακέτα με προορισμό τον mobile node θα πηγαίνουν στο δίκτυο X, ενώ ο mobile node βρίσκεται στο δίκτυο Y.
2. Στην περίπτωση που ο mobile node αλλάξει την IP διεύθυνση με network prefix Y, τότε οι nodes που θα ήθελαν να επικοινωνήσουν με τον mobile node δεν θα γνωρίζουν την νέα του διεύθυνση.

Η διένεξη λοιπόν που παρουσιάζεται παραπάνω, πρέπει να αντιμετωπισθεί στο επίπεδο δικτύου (network layer) και όχι π.χ. στο επίπεδο μεταφοράς (TCP,UDP). Απομονώνοντας το πρόβλημα στο network layer, δεν γίνεται αντιληπτό στα παραπάνω επίπεδα ότι έχει αλλάξει το σημείο πρόσβασης του mobile node.

Οι λύσεις που έχουν προταθεί είναι αρκετές. Στην παρούσα εργασία θα εξετάσουμε κυρίως το Mobile IP.

2 Mobile IP

Το Mobile IP [1], [2] είναι μία επέκταση του γνωστού IP πρωτόκολλου. Σκοπός του Mobile IP είναι να παρέχει τους κατάλληλους μηχανισμούς έτσι ώστε να επιτρέπει στο κινητό σταθμό να κινείται σε διάφορα δίκτυα διατηρώντας τις υπάρχουσες συνδέσεις του. Το Mobile IP υποβλήθηκε ως προτεινόμενο πρότυπο στο Internet Engineering Steering Group (IESG) τον Οκτώβριο του 1996.

Πριν όμως προχωρήσουμε στην ανάλυση του Mobile IP, κρίνεται σκόπιμο να επεξηγήσουμε τις νέες έννοιες και λειτουργικές οντότητες, όπως αυτές περιγράφονται στις προδιαγραφές του.

- **Mobile Node –** Ένας host ή router ο οποίος αλλάζει το σημείο σύνδεσής του από ένα δίκτυο σε ένα άλλο χωρίς να αλλάζει την IP διεύθυνση του. Ένας mobile node μπορεί να συνεχίσει να επικοινωνεί με άλλους κόμβους στο Διαδίκτυο κάνοντας χρήση της σταθερής IP διεύθυνσής του.

- **Home Agent –** Ένας δρομολογητής στο οικείο δίκτυο (Home network) ο οποίος διανέμει τα πακέτα στον απομακρυσμένο mobile node. Επίσης ο Home Agent αναλαμβάνει να διατηρεί πληροφορίες για την τρέχουσα θέση του mobile node.

- **Foreign Agent –** Ένας δρομολογητής στο ξένο δίκτυο στο οποίο βρίσκεται ο mobile node.Ο Foreign Agent συνεργάζεται με τον Home Agent για να ολοκληρώσει τη παράδοση των πακέτων προς τον mobile node όσο αυτός βρίσκεται στο ξένο δίκτυο.

- **Agent Advertisement –** Η διαδικασία με την οποία ο Foreign Agent γνωστοποιεί την παρουσία του. Αυτό γίνεται με ένα ειδικό μήνυμα το οποίο στην ουσία

αποτελείται από ένα router advertisement με μια ειδική προέκταση.

- **Care-of-Address –** Είναι το τελικό σημείο του τούνελ από τον Home Agent προς τον mobile node όσο ο τελευταίος βρίσκεται εκτός του home network. Είναι στην ουσία η νέα διεύθυνση στην οποία καταλήγουν τα ενθυλακωμένα (encapsulated) πακέτα που στέλνει ο Home Agent. Υπάρχουν δύο ειδών Care-of-Address:

    - **Foreign Agent Care-of-Address –** Είναι η διεύθυνση του Foreign Agent στον οποίο ο Mobile Node είναι καταχωρημένος. Αξίζει να σημειωθεί ότι την αποθυλάκωση των ενθυλακωμένων πακέτων από τον Home Agent προς τον Mobile Node την αναλαμβάνει ο Foreign Agent.

    - **Collocated Care-of-Address –** Πρόκειται για τη νέα διεύθυνση την οποία αποκτά δυναμικά ο Mobile Node (μέσω μηχανισμών DHCP ή PPP) στο ξένο δίκτυο. Σε αυτήν τη περίπτωση την αποθυλάκωση των πακέτων αναλαμβάνει ο ίδιος ο Mobile Node.

- **Correspondent node –** Είναι ο κόμβος με τον οποίο επικοινωνεί ο Mobile node.Ο Correspondent node μπορεί να είναι είτε σταθερός είτε κινητός.

- **Foreign Network -** Είναι όλα τα δίκτυα εκτός του Home Network.

- **Home Address -** Είναι η IP διεύθυνση η οποία αποδίδεται στο Mobile Node για μεγάλο χρονικό διάστημα. Η home address παραμένει αμετάβλητη ανεξάρτητα του σημείου πρόσβαση του Mobile Node.

- **Home Network –** Ένα δίκτυο (πιθανών και ιδεατό (virtual)) του οποίου η διεύθυνση δικτύου ταυτίζεται με την διεύθυνση δικτύου της home address. Εδώ

να σημειώσουμε ότι, τα πακέτα που προορίζονται προς την home address του mobile node δρομολογούνται βάσει των γνωστών IP μηχανισμών δρομολόγησης και άρα δρομολογούνται προς το home network.

- **Tunnel –** Το μονοπάτι (path) που διανύει ένα ενθυλακωμένο πακέτο από το home network προς το mobile node.

2.1 Ανάλυση του Mobile IP

Παρακάτω, ακολουθεί μία περιγραφή της λειτουργίας του Mobile IP.

Οι Agents (Foreign ή Home) διαφημίζουν τη παρουσία τους στο δίκτυο εκπέμποντας περιοδικά μηνύματα *Agent Advertisements.* Ο mobile node έχει τη δυνατότητα να ζητήσει *Agent Advertisement* εκπέμποντας μηνύματα του τύπου *Agent Solicitation.* Εδώ να σημειώσουμε ότι ο mobile node είναι σε θέση να λαμβάνει συνέχεια μηνύματα Agent Advertisements.

Στη συνέχεια, αφού ο mobile node έχει λάβει ένα Agent Advertisement από τον agent, ελέγχει το περιεχόμενο του μηνύματος για να κρίνει εάν βρίσκεται στο οικείο ή σε ξένο δίκτυο.

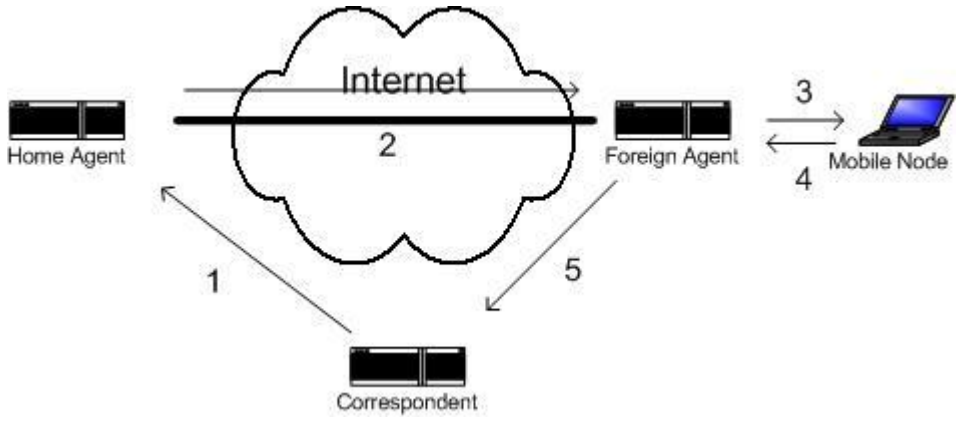
Εάν βρίσκεται στο οικείο δίκτυο (home network) τότε ενεργεί σαν ένας σταθερός κόμβος αυτού του δικτύου, κάνοντας χρήση της σταθερής διεύθυνσής του (Home Address).

Εάν ο mobile node αντιληφθεί ότι βρίσκεται σε ξένο δίκτυο, τότε ο mobile node πρέπει να λάβει μια Care-of-address ή μια collocated Care-of-address. Η διαδικασία απόκτησης των διευθύνσεων περιγράφεται παρακάτω.

Αφού λοιπόν ο Mobile Node λάβει είτε την Care-of-address είτε την collocated Care-of-address, στη συνέχεια πρέπει να την γνωστοποιήσει (register) στον home agent. Αυτό γίνεται κάνοντας χρήση των Registration Messages.

Ο mobile node αποστέλλει ένα μήνυμα Registration Request στον home agent μέσω του foreign agent. Στη συνέχεια ο home agent θα πιστοποιήσει (authentication) το μήνυμα και θα απαντήσει στον mobile node, μέσω πάντα του foreign agent, με ένα μήνυμα Registration Reply. Εάν το Registration Reply είναι καταφατικό τότε ο mobile node είναι πλήρως καταχωρημένος στον home agent του και είναι έτοιμος να στείλει και να λάβει πακέτα.

Στη συνέχεια, με τη βοήθεια του σχήματος 5, περιγράφεται η δρομολόγηση των πακέτων από και προς τον mobile node, από τη στιγμή που ο mobile node έχει καταχωρηθεί στον home agent.



Σχήμα 5

Το πακέτο από τον correspondent host που προορίζεται για τον mobile node, φθάνει στον home agent μέσω των καθιερωμένων IP δρομολογητικών μηχανισμών. Στη συνέχεια το πακέτο ¨αναχαιτίζεται¨ από τον home agent, ενθυλακώνεται και δρομολογείται μέσω κάποιων μηχανισμών tunneling (οι οποίοι περιγράφονται αναλυτικά παρακάτω) προς την Care-of-address. Όταν το πακέτο φθάσει στην Care-of-address, αποθυλακώνεται και παραδίδεται στον mobile node. Η αντίστροφη διαδικασία, δηλαδή η αποστολή πακέτων από τον mobile node προς τον correspondent node γίνεται με δύο τρόπους.

- Reverse Tunneling

  Σύμφωνα με αυτήν τη διαδικασία, τα πακέτα που αποστέλλονται από τον mobile node, ενθυλακώνονται είτε από την foreign agent είτε από τον ίδιο τον mobile node, στέλνονται home agent, αποθυλακώνονται και στη συνέχεια αποστέλλονται προς τον correspondent node. Στην ουσία είναι η αντίστροφη

διαδικασία για τα εισερχόμενα πακέτα.

- Triangle Routing

  Σε αυτήν τη περίπτωση, τα εξερχόμενα πακέτα από τον mobile node, αποστέλλονται κατευθείαν από τον ίδιο προς τον correspondent node μέσω του foreign agent χωρίς να περάσουν από τον home agent.

2.2 Ανάλυση Επιμέρους Λειτουργιών

Agent Advertisement και Agent Solicitation

Η διαδικασία της ανίχνευσης των agent είναι παρόμοια με αυτήν που χρησιμοποιείται από τους κόμβους στο Διαδίκτυο για την αναζήτηση δρομολογητών, οι οποίοι τρέχουν το Internet Control Message Protocol (ICMP) Router Discovery. Η βασική λειτουργία του, έχει να κάνει με την περιοδική εκπομπή διαφημιστικών μηνυμάτων από τους δρομολογητές στα υποδίκτυα με τα οποία συνδέονται.

Αξίζει να σημειωθεί, ότι η ομάδα εργασίας του Mobile IP, αποφάσισε να χρησιμοποιήσει αυτούσιο το RFC 1256 (που περιγράφει το ICMP Router Discovery) και απλώς να προσθέσει κάποια προέκταση στα καθιερωμένα ICMP μηνύματα.

Στην παρούσα εργασία, θα αναλύσουμε μόνο την προέκταση αυτή (το λεγόμενο mobility agent extension). Το mobility agent extension, πληροφορεί τον mobile node καταρχήν για το αν είναι διαθέσιμος εκείνη τη στιγμή ο agent, αν είναι home ή foreign agent, τι είδους αλγόριθμο ενθυλάκωσης εφαρμόζει (π.χ. minimal ή GRE encapsulation), αν εφαρμόζει Van Jacobson συμπίεση επικεφαλίδας κλπ. Επίσης, στο Agent Advertisement μήνυμα, εμπεριέχονται και οι Care-of-Addresses που διαθέτει ο foreign agent. Στο σχήμα 6, φαίνεται το mobility agent extension.

Σχήμα 6

Όσον αφορά το Agent Solicitation μήνυμα, αυτό αποστέλλεται από τον mobile node με σκοπό να ζητήσει από τον agent ένα agent advertisement μήνυμα. Η μορφή των agent solicitation μηνυμάτων, είναι η ίδια με τα ICMP Router Solicitation μηνύματα.

Care-of-Address

Στις προδιαγραφές του Mobile IP περιγράφονται δύο τρόποι για την απόκτηση προσωρινών διευθύνσεων (Care-of-Addresses). Είναι η foreign Care-of-Address και η collocated Care-of-Address.

- Η foreign Care-of-Address είναι η προσωρινή διεύθυνση που παρέχεται από τον foreign agent στον mobile node, μέσω των agent advertisement μηνυμάτων όπως προαναφέραμε. Η διεύθυνση αυτή, είναι μία IP διεύθυνση του foreign network. Σε αυτή την περίπτωση, το τελικό σημείο του tunnel είναι ο foreign agent ο οποίος και αναλαμβάνει την αποθυλάκωση των πακέτων που αποστέλλει ο home agent και αποστέλλει το εσωτερικό πακέτο στον mobile node. Αυτός ο τρόπος διευθυνσιοδότησης προτιμάται, διότι έτσι σε μία Care-of-Address μπορούμε να συσχετίσουμε πολλούς mobile nodes.
- Η collocated Care-of-Address είναι μία τοπική IP διεύθυνση του foreign

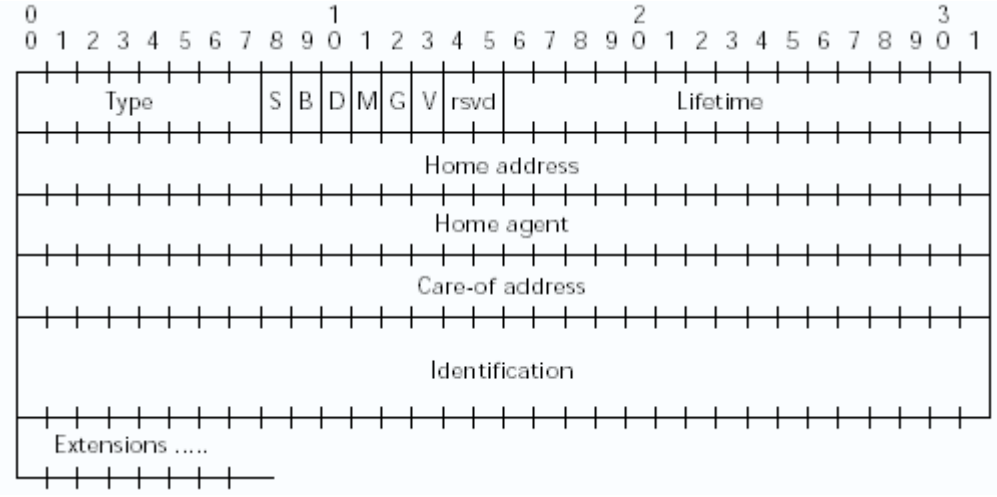
network την οποία αποκτά δυναμικά ο mobile node μέσω PPP ή DHCP πρωτοκόλλων. Όταν γίνεται χρήση της collocated Care-of –Address, το τελικό σημείου του tunnel είναι ο ίδιος ο mobile node ο οποίος και αναλαμβάνει την αποθυλάκωση των πακέτων. Το πλεονέκτημα αυτού του τρόπου λειτουργίας είναι ότι αποφορτίζεται ο foreign agent από την αποθυλάκωση των πακέτων. Το μειονέκτημα όμως είναι ότι είναι δύσκολο κάθε δίκτυο να διαθέτει έναν αριθμό IP διευθύνσεων για mobile nodes οι οποίοι ενδεχομένως θα επισκεφθούν το δίκτυο, εάν αναλογιστούμε και την έλλειψη IP διευθύνσεων στο IPv4, αλλά και το ότι κάθε collocated Care-of-Address αντιστοιχεί σε έναν mobile node.

Registration Process

Όπως αναφέραμε παραπάνω, η διαδικασία του Registration αποτελείται από την αποστολή από τον mobile node προς τον home agent ενός *Registration Request* μηνύματος και εν συνεχεία από την αποστολή από τον home agent προς τον mobile node ενός *Registration Reply* μηνύματος. Και τα δύο μηνύματα είναι UDP πακέτα. Ας δούμε όμως αναλυτικότερα τα δύο αυτά μηνύματα.

Registration Request

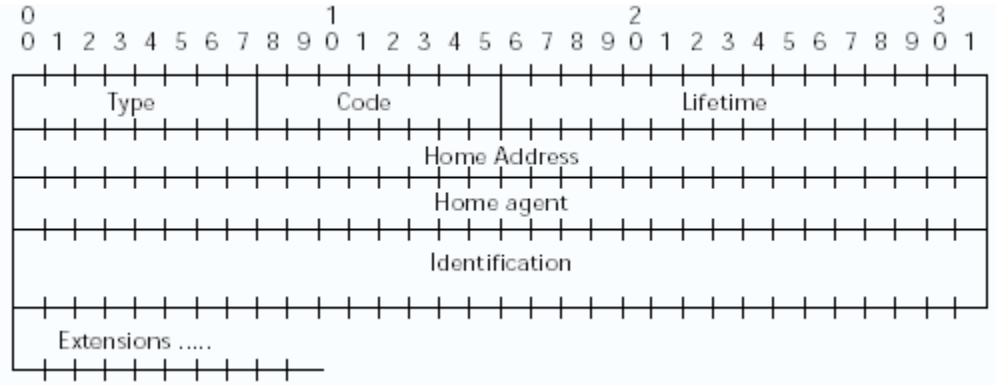
Στο σχήμα 7 απεικονίζεται η δομή του Registration Request μηνύματος. Με το εν λόγω μήνυμα, ο mobile node πληροφορεί την home agent για την τρέχουσα Care-of-address που του έχει αποδοθεί από τον foreign agent, το χρονικό διάστημα για το οποίο ο mobile node θέλει να είναι registered και τυχόν ειδικές υπηρεσίες που προσφέρει ο foreign agent. Επίσης μέσω του Registration Request μηνύματος, πληροφορείται ο home agent για το αν χρησιμοποιείται Van Jacobson συμπίεση επικεφαλίδας και τι είδους tunneling αλγόριθμος χρησιμοποιείται.



Σχήμα 7

Registration Reply

Το Registration Reply είναι η απάντηση του home agent στο Registration Request του mobile node. Στο σχήμα 8, φαίνεται το εν λόγω μήνυμα. Στο Registration Reply αναφέρεται καταρχήν εάν γίνεται δεκτό το αίτημα του mobile node για registration. Εάν η απάντηση είναι καταφατική, τότε στο μήνυμα αναφέρεται το χρονικό διάστημα για το οποίο ο mobile node θα είναι registered.



Σχήμα 8

Εάν η απάντηση είναι αρνητική, τότε αναφέρονται και οι λόγοι για τους οποίος

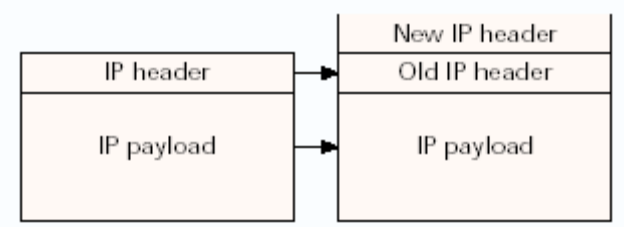
δεν είναι εφικτό το registration. Εδώ αξίζει να σημειωθεί ότι μια αρνητική απάντηση μπορεί να οφείλεται και στον home agent αλλά και στον foreign agent.

Εάν ο home agent αρνηθεί το registration αυτό μπορεί να οφείλεται σε μη πιστοποίηση του mobile node, σε λανθασμένη συμπλήρωση του Registration Request από τον mobile node ή σε ανεπάρκεια πόρων του home agent για καταχώρηση του mobile node την δεδομένη στιγμή.

Εάν ο foreign agent αρνηθεί το registration, αυτό μπορεί να οφείλεται στην αδυναμία του foreign agent να επικοινωνήσει με τον home agent ώστε να του αποστείλει το Registration Request ή και σε ελλιπή συμπλήρωση του Registration Reply από τον home agent.

Tunneling

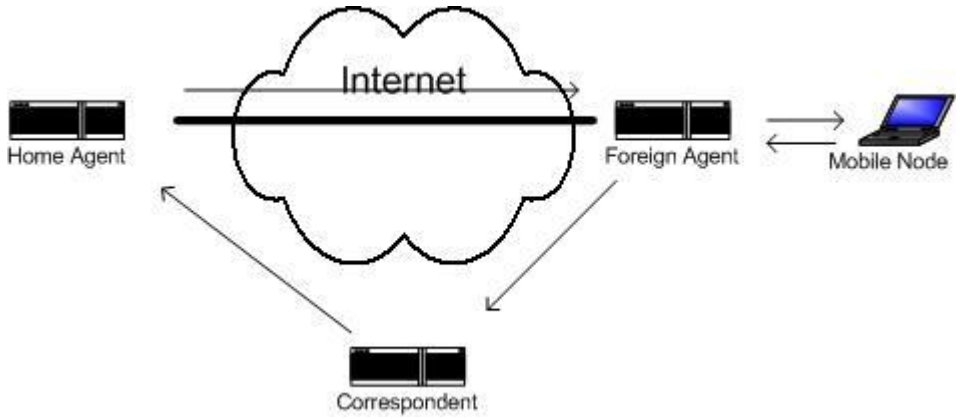
Με τον όρο tunneling εννοούμε την διαδικασία κατά την οποία ένα πακέτο τοποθετείται μέσα σε ένα άλλο πακέτο (ενθυλακώνεται) και στην συνέχεια αποστέλλεται στον τελικό προορισμό του. Πιο συγκεκριμένα, ο home agent, μετά από μία επιτυχή καταχώρηση του mobile node, θα «προσελκύσει» τα πακέτα που προορίζονται για τον mobile node και μέσω της διαδικασίας tunneling θα τα αποστείλει, ενθυλακωμένα πλέον, προς τον mobile node. Το tunneling μπορεί να πραγματοποιηθεί κάνοντας χρήση διαφόρων αλγορίθμων, αλλά ο προκαθορισμένος είναι ο IP-within-IP encapsulation. Στο σχήμα 9, φαίνεται πως ενθυλακώνεται ένα IP πακέτο με την προσθήκη μιας IP επικεφαλίδας (η tunnel επικεφαλίδα). Εναλλακτικά, μπορεί να γίνει χρήση του αλγορίθμου minimal encapsulation, εάν συμφωνήσουν ο mobile node, ο home agent και ο foreign agent.



Σχήμα 9

2.3 Θέματα Δρομολόγησης στο Mobile IP

Ο μηχανισμός δρομολόγησης που περιγράφεται στο σχήμα 10, ονομάζεται **triangle routing** (τριγωνική δρομολόγηση). Δηλαδή, ο mobile node επικοινωνεί απευθείας με τον correspondent node μέσω του foreign agent, ενώ ο correspondent node επικοινωνεί με τον mobile node μέσω του home agent και γενικότερα όλης της διαδικασίας που περιγράψαμε παραπάνω. Το triangle routing είναι ένα από τα σημαντικότερα προβλήματα που παρουσιάζει το Mobile IP διότι υπάρχει ένα επιπλέον μονοπάτι ανάμεσα στον home agent και τον mobile node. Σε περιπτώσεις που ο mobile node βρίσκεται τον περισσότερο χρόνο στο home network και περιστασιακά μετακινείται σε γειτονικά δίκτυα, το triangle routing δεν αποτελεί ιδιαίτερο πρόβλημα. Εν τούτης, ο mobile node μπορεί να βρίσκεται μακριά από το home network και σε μερικές περιπτώσεις πολύ κοντά με τον correspondent node. Σε μία τέτοια περίπτωση, η τριγωνική δρομολόγηση αποτελεί σημαντικότατο ζήτημα, διότι τα πακέτα διανύουν μεγάλη διαδρομή μέχρι να φθάσουν στον mobile node με αποτέλεσμα να παρουσιάζεται μεγάλη καθυστέρηση στην επικοινωνία mobile node με correspondent node.



Σχήμα 10

Όπως αναφέραμε παραπάνω, ο mobile node αποστέλλει τα πακέτα προς τον correspondent node απευθείας χωρίς να είναι ανάγκη να περάσουν μέσω του home agent. Στην περίπτωση αυτή, η διεύθυνση πηγής των πακέτων είναι η home address του mobile node και η διεύθυνση προορισμού είναι η IP address του correspondent

node. Αυτή η μορφή επικοινωνίας όμως δεν είναι πάντα εφικτή. Αυτό συμβαίνει διότι υπάρχει η πιθανότητα κάποιοι routers να κάνουν χρήση του πρωτοκόλλου Network Ingress Filtering. Δηλαδή οι routers να απορρίπτουν πακέτα των οποίων η διεύθυνση πηγής δεν είναι τοπολογικά σωστή.

Σε ένα τέτοιο σενάριο, ο mobile node όταν βρίσκεται σε ένα foreign network, δεν μπορεί να χρησιμοποιήσει την home address ως διεύθυνση πηγής για να στείλει πακέτα στον correspondent node. Για τη λύση αυτού του προβλήματος έχει προταθεί η τεχνική του reverse tunneling, κατά την οποία, τα πακέτα που προορίζονται για τον correspondent node, ο mobile node μέσω τεχνικών tunneling τα αποστέλλει προς τον home agent και εκείνος με τη σειρά του, αφού τα αποθυλακώσει, τα στέλνει στον correspondent node. Η διαδικασία αυτή δεν παρουσιάζει προβλήματα, όσον αφορά το ingress filtering, αφού η διευθυνσιοδότηση που χρησιμοποιεί τοπολογικά είναι σωστή. Το μειονέκτημα όμως που παρουσιάζει είναι ότι εμφανίζει την τριγωνική δρομολόγηση στην αντίθετη κατεύθυνση.

Για παραπάνω προβλήματα δρομολόγησης, έχουν προταθεί διάφορες λύσεις, οι οποίες ονομάζονται **route optimization** τεχνικές. Παρακάτω αναλύονται οι τεχνικές αυτές .

Standard Route Optimization

Η τεχνική Standard Route Optimization [4] χρησιμοποιείται για την βελτιστοποίηση της δρομολόγησης των πακέτων από τον correspondent node προς τον mobile node. Εδώ βέβαια, γίνεται η υπόθεση ότι η απευθείας δρομολόγηση των πακέτων από τον mobile node προς τον correspondent node είναι εφικτή, δηλαδή δεν έχουμε ingress filtering.

Η βελτιστοποίηση του δρομολογίου mobile node προς correspondent node, επιτυγχάνεται με τη χρήση **bindings** (δεσμών) από τον τελευταίο. Ο correspondent node δηλαδή γνωρίζει την Care-of-address του mobile node οπότε στέλνει πλέον τα πακέτα ενθυλακωμένα προς τον ίδιο τον mobile node. Τα bindings πρέπει να

παρέχονται σε όλους τους correspondent nodes που θέλουν να επικοινωνήσουν με τον mobile node. Άπαξ ο correspondent node δημιουργήσει ένα binding για έναν συγκεκριμένο mobile node, αυτό πρέπει να ανανεώνεται κάθε φορά που ο mobile node αλλάζει Care-of-address έτσι ώστε να διασφαλίζεται η σωστή δρομολόγηση των πακέτων. Το σημαντικότερο ζήτημα στην Standard Route Optimization τεχνική, είναι η ανανέωση των bindings. Για το σκοπό αυτό έχουν καθορισθεί τέσσερις τύποι μηνυμάτων.

Το *binding warning* μήνυμα αποστέλλεται προς τον home agent για να τον ενημερώσει ότι ο Correspondent node δεν γνωρίζει την care-of-Address ενός συγκεκριμένου mobile node. Ο correspondent node επίσης, μπορεί να στείλει ένα *binding request* μήνυμα, προς τον home agent, για να ζητήσει binding. Τα *bindings update* μηνύματα, χρησιμοποιούνται για την αποστολή της νέας Care-of-Address του mobile node, προς τον correspondent node. Ο home agent είναι υπεύθυνος για την αποστολή των bindings update μηνυμάτων προς τους correspondent nodes.

Όταν ο correspondent node επικοινωνεί πρώτη φορά με τον mobile node μέσω του home agent, ο τελευταίος αυτόματα στέλνει ένα binding update μήνυμα στον correspondent node για να τον ενημερώσει για την Care-of-Address του mobile node. Να σημειώσουμε εδώ, ότι το binding update μήνυμα εμπεριέχει και το χρονικό διάστημα για το οποίο η Care-of-Address θα είναι έγκυρη.

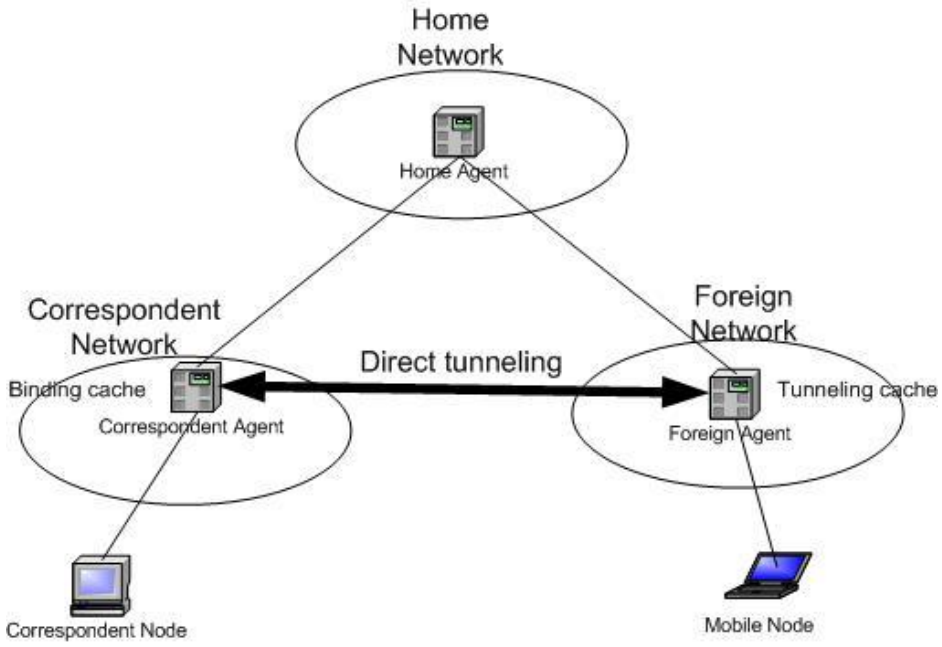
Η Standard Route Optimization τεχνική, δίνει τη δυνατότητα στον mobile node να αποστείλει ο ίδιος binding update μήνυμα προς τον correspondent node και αυτό γίνεται μόλις ο mobile node αλλάξει Care-of-Address για λόγους ταχύτητας. Σε αυτήν την περίπτωση, ο mobile node στέλνει ένα *binding acknowledgement* μήνυμα για να ενημερωθεί εάν ο correspondent node έχει λάβει τη νέα Care-of-Address. Όπως καταλαβαίνουμε, ο home agent δεν χρειάζεται να στείλει binding acknowledgement μήνυμα διότι εάν ο πρώτος δέχεται πακέτα που προορίζονται για τον mobile node, τότε καταλαβαίνει ότι ο correspondent node δεν έχει ενημερωθεί με το binding update μήνυμα.

Το μειονέκτημα της Standard Route Optimization τεχνικής, είναι ότι απαιτεί από

τους correspondent nodes να επιφορτιστούν με παραπάνω έργο, πράγμα το οποίο έρχεται σε αντίθεση με μία βασική αρχή του Mobile IP σύμφωνα με την οποία οι κόμβοι που επικοινωνούν με τον mobile node δεν αντιλαμβάνονται την κίνησή του. Εδώ όμως είδαμε ότι οι correspondent nodes πρέπει να διατηρούν μία binding cache αλλά επίσης και να έχουν τη δυνατότητα να αποστέλλουν τα πακέτα προς τον mobile node ενθυλακωμένα. Υπάρχει όμως η πιθανότητα πολλοί correspondent nodes να μην έχουν υιοθετήσει το Standard Route Optimization πρωτόκολλο. Σε αυτήν τη περίπτωση τότε, αν ο home agent έχει στείλει binding update μήνυμα προς τον correspondent node αλλά εκείνος συνεχίζει να στέλνει τα πακέτα προς τον home agent, ο τελευταίος καταλαβαίνει ότι ο correspondent node δεν κάνει χρήση του εν λόγω πρωτόκολλου. Οπότε ακολουθείται η παραδοσιακή διαδικασία δρομολόγησης.

Bi-directional Route optimization

Η Standard Route Optimization τεχνική υποθέτει ότι δεν γίνεται χρήση ingress filtering [3] από τους routers. Οπότε, αρκείται στην βελτιστοποίηση δρομολογίου, μόνο όσων αφορά την διαδρομή correspondent node προς mobile node. Εάν όμως λάβουμε υπ όψιν μας το ingress filtering, τα πακέτα από τον mobile node προς τον correspondent node δεν μπορούν να σταλθούν απευθείας. Όπως είδαμε παραπάνω, μία λύση είναι να στέλνει τα πακέτα ο mobile node μέσω της τεχνικής reverse tunneling. Έτσι όμως εμφανίζεται το πρόβλημα της τριγωνικής δρομολόγησης στην αντίθετη κατεύθυνση. Για την λύση αυτού του προβλήματος, έχει προταθεί η τεχνική **bi-directional route optimization** [7]. Στο σχήμα 12 απεικονίζεται η βασική αρχιτεκτονική του bi-directional route optimization.



Σχήμα 11

Η συγκεκριμένη αρχιτεκτονική εισάγει μία νέα οντότητα, τον correspondent agent ο οποίος διαχειρίζεται τις binding caches και επίσης αναλαμβάνει την ενθυλάκωση των πακέτων. Ο correspondent agent συνεργάζεται με πολλούς correspondent nodes.

Η βασική καινοτομία είναι ότι ο foreign agent αναλαμβάνει τη τήρηση της tunneling cache. Η tunneling cache είναι στην ουσία μία λίστα με τους correspondent nodes οι οποίοι υποστηρίζουν το bi-directional route optimization. Σε αντίθεση με την binding cache, η τήρηση της tunneling cache είναι πολύ ευκολότερη διότι δεν υπάρχει η ανάγκη για ανανέωση των εγγραφών. Παρακάτω εξηγούμε την λειτουργία της συγκεκριμένης τεχνικής.

Εάν υπάρχει καταχώρηση στη tunneling cache για ένα correspondent node, τότε ο mobile node μπορεί να στείλει απευθείας ενθυλακωμένα πακέτα προς τον correspondent node. Η διεύθυνση πηγής του εξωτερικού πακέτου του ενθυλακωμένου πακέτου, είναι η Care-of-Address του mobile node η οποία είναι τοπολογικά σωστή οπότε δεν θα έχουμε απόρριψη πακέτων λόγω ingress filtering.

Από την πλευρά του correspondent agent τώρα, εάν δεν υπάρχει μια binding εγγραφή για έναν συγκεκριμένο mobile node η αποστολή των πακέτων προς αυτόν θα

γίνει μέσω του home agent. Αυτός με τη σειρά του, μόλις λάβει το πρώτο πακέτο, θα στείλει ένα binding update μήνυμα προς τον correspondent agent και έτσι δημιουργείται μια binding εγγραφή στον τελευταίο. Από την στιγμή που έχει δημιουργηθεί η binding εγγραφή ο correspondent agent θα στέλνει tunneled τα πακέτα απευθείας προς τον mobile node. Μόλις ο foreign agent με τη σειρά του λάβει το πρώτο ενθυλακωμένο πακέτο, αυτόματα δημιουργεί μία εγγραφή στην tunneling cache για τον συγκεκριμένο correspondent agent οπότε μπορεί να στέλνει tunneled πακέτα απευθείας στον correspondent agent.

Διαφορετικά, αν δεν υπάρχει εγγραφή στην tunneling cache για έναν συγκεκριμένο correspondent agent, ο foreign agent κάνοντας χρήση της reverse tunneling τεχνικής θα αποστείλει τα πακέτα προς τον home agent.

Όπως φαίνεται καθαρά, το μειονέκτημα αυτής της τεχνικής είναι ότι εισάγει μία νέα οντότητα, τον correspondent agent και απαιτεί από τον foreign agent να υποστηρίζει επιπλέον λειτουργίες. Πρέπει δηλαδή οι correspondent agents να τηρούν μία binding cache και να ενθυλακώνουν πακέτα, και οι foreign agents να τηρούν την tunneling cache.

Ωστόσο, τα πλεονεκτήματα που προσφέρει είναι σημαντικότερα. Καταρχήν, μειώνεται η κίνηση από και προς τον home agent. Επίσης βελτιώνεται το δρομολόγιο αμφίπλευρα με άμεσο αντίκτυπο στην μείωση της καθυστέρησης στην επικοινωνία correspondent node με mobile node.

3 Mobile IP και Handover

Συνοψίζοντας ως εδώ, έχουμε δει πως προσεγγίζει το πρόβλημα της κινητικότητας το πρωτόκολλο Mobile IP. Είδαμε ότι η βασική αρχή λειτουργίας του είναι η τήρηση δύο διευθύνσεων, μίας σταθερής (της home address) και μίας προσωρινής (της Care-of-Address). Είδαμε πως επιτυγχάνεται η επικοινωνία μεταξύ mobile node και correspondent node και πως προσεγγίζονται τα διάφορα προβλήματα δρομολόγησης που παρουσιάζονται.

Μέχρι στιγμής όμως, δεν έχουμε μιλήσει για την περίπτωση που ο mobile node κινείται από ένα δίκτυο σε ένα άλλο. Δηλαδή, πως αντιμετωπίζει το Mobile IP την κίνηση του mobile node και τι επιπτώσεις έχει αυτό στην επικοινωνία του mobile node με τον correspondent node.

Εδώ να σημειώσουμε ότι η αντιμετώπιση της κίνησης του mobile node από τα διάφορα πρωτόκολλα κινητικότητας, είναι ο κύριος παράγοντας που χαρακτηρίζει την αποδοτικότητα του εκάστοτε πρωτοκόλλου.

Όπως καταλαβαίνουμε, αφού αναφερόμαστε στην κίνηση του mobile node και εν συνεπεία στην αλλαγή του σημείου πρόσβασης του mobile node με το Internet, μιλάμε στην ουσία για το handover.

3.1 Handover

Με τον όρο handover (μεταπομπή) αναφερόμαστε στην διαδικασία αλλαγής του σημείου σύνδεσης ενός mobile node κατά την κίνηση του από ένα (υπό-)δίκτυο σε ένα άλλο.

Η διαδικασία του handover αποτελείται από δύο στάδια.

I. Την εκκίνηση (initiation), δηλαδή το εάν πρέπει να γίνει και πότε να

γίνει το handover,άρα μιλάμε για **handover initiation αλγόριθμους** και

II. την υλοποίηση (execution) του handover, δηλαδή με ποιόν τρόπο θα γίνει, όπου εδώ αναφέρονται ως **handover execution αλγόριθμοι**.

Αξίζει να σημειωθεί εδώ, ότι όσον αφορά το Mobile IP, έχουν αναπτυχθεί πολύ περισσότεροι αλγόριθμοι για την διαδικασία υλοποίησης του handover σε σχέση με τη διαδικασία εκκίνησης. Παρακάτω αναλύουμε τους αλγόριθμους όσον αφορά και τα δύο στάδια του handover.

3.2 Είδη handover

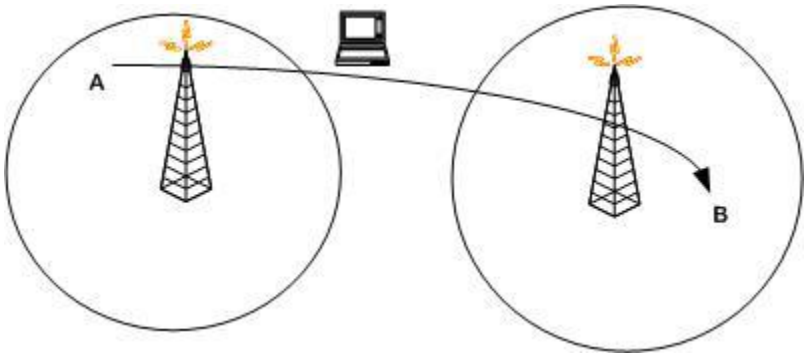
Για να παρέχονται από το δίκτυο προς τον mobile node αδιάλειπτες υπηρεσίες και συνεχή επικοινωνία (για παράδειγμα IP τηλεφωνία) τα handover πρέπει να έχουν υψηλή απόδοση, δηλαδή να πληρούν τις παρακάτω προϋποθέσεις:

1. Τα handover πρέπει να ολοκληρώνονται γρήγορα δηλαδή να παρουσιάζουν μικρό handover latency.
2. Τα handover πρέπει να ελαχιστοποιούν (τυπικά να μηδενίσουν) τις απώλειες πακέτων (packet loss).
3. Κατά την διαδικασία του handover να μην «φορτώνεται» το δίκτυο με υπερβολική κίνηση ελέγχου.

Βασιζόμενοι στις παραπάνω «μετρήσεις απόδοσης» οι handover αλγόριθμοι διαχωρίζονται σε τρεις κατηγορίες:

1. Fast handover: Όταν κατά την διαδικασία του handover παρουσιάζεται μικρή διακοπή μεταξύ των χρονικών στιγμών της αποσύνδεσης από το ένα point of attachment και της σύνδεσης στο νέο point of attachment.
2. Smooth handover: Όταν η απώλεια πακέτων είναι μηδαμινή αλλά παρόλα αυτά η διάρκεια της διακοπής της επικοινωνίας είναι σημαντική.
3. Seamless handover: Συνδυασμός fast και smooth handover.

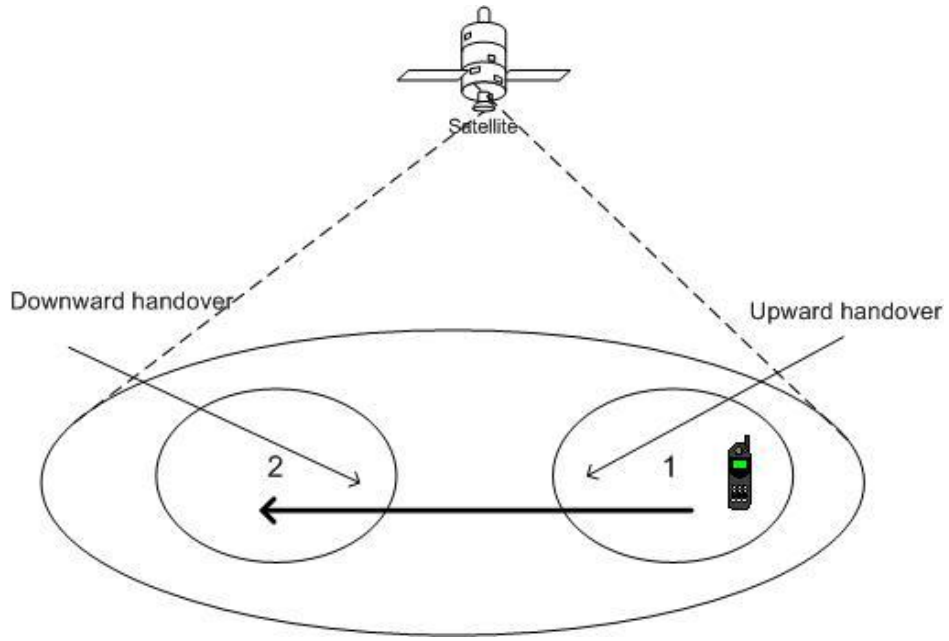
Όπως φαίνεται, το seamless handover είναι η βέλτιστη λύση. Ένα τέτοιο handover όμως δεν είναι πάντα εφικτό. Και αυτό διότι πολλές φορές μπορεί να μην έχουμε αλληλοεπικάλυψη των κυψελών, όπως φαίνεται και στο σχήμα 12.



Σχήμα 12

Ένα άλλο κριτήριο διαχωρισμού του handover είναι η ιεραρχία των κυψελών στις οποίες λαμβάνει χώρα το handover. Όταν το handover γίνεται μεταξύ κυψελών ίδιας ιεραρχίας, τότε αναφέρεται ως horizontal handover. Όταν όμως έχουμε κυψέλες διαφορετικής ιεραρχίας, τότε έχουμε vertical handover. Ειδικότερα, εάν έχουμε κίνηση από χαμηλότερη κυψέλη σε υψηλότερη, τότε μιλάμε για upward handover ενώ στην αντίθετη περίπτωση έχουμε downward handover.

Στο σχήμα 3-2, βλέπουμε ένα παράδειγμα downward και upward handover.

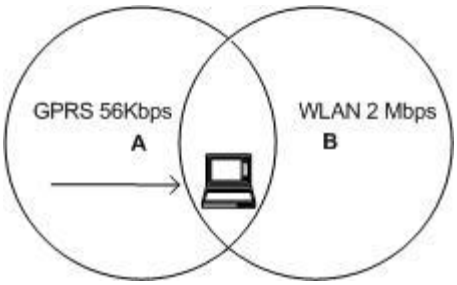


Σχήμα 13

Στο σχήμα 13 απεικονίζεται η μακροκυψέλη από τον δορυφόρο όπου καλύπτει γεωγραφικά τις δύο μικρότερες κυψέλες 1 και 2. Στο σενάριο αυτό, ο mobile node θα κινηθεί από την κυψέλη 1 προς την κυψέλη 2. Όπως βλέπουμε οι κυψέλες 1 και 2 δεν αλληλεπικαλύπτονται, οπότε ο κινητός σταθμός θα εξυπηρετηθεί από τον δορυφόρο στην περιοχή όπου δεν υπάρχει κάλυψη από επίγειους σταθμούς. Κατά την έξοδο του mobile node από την κυψέλη 1 εμφανίζεται το upward handover. Εδώ το handover latency πρέπει να είναι όσο το δυνατόν μικρό, διότι μέχρι τη χρονική στιγμή που θα τον «σερβίρει» ο δορυφόρος, δεν έχει κάλυψη από τον σταθμό 1.

Κατά την είσοδο του mobile node στην κυψέλη 2, όπου εκεί εμφανίζεται το downward handover, υπάρχει μια ανοχή, σε ότι αφορά το handover latency, και αυτό διότι μέχρι να ολοκληρωθεί το handover, ο mobile node μπορεί να εξυπηρετείται από τον δορυφόρο.

Στη συνέχεια, ας δούμε ένα παράδειγμα, όπου φαίνεται η σπουδαιότητα του handover initiation αλγόριθμου.



Σχήμα 14

Στο σενάριο, που απεικονίζεται στο σχήμα 14, έχουμε δύο κυψέλες που αλληλεπικαλύπτονται. Την Α και τη Β. Στη κυψέλη Α παρέχεται GPRS πρόσβαση με data rate 56 Kbps, ενώ στη κυψέλη 2 έχουμε ένα WLAN με data rate 2 Mbps. Ο mobile node βρισκόταν στη κυψέλη Α και στη συνέχεια μετακινήθηκε στην περιοχή αλληλοεπικάλυψης των δύο κυψελών. Εάν ο handover initiation αλγόριθμος αποσκοπεί αποκλειστικά στην διατήρηση της σύνδεσης με το Internet, δεν εκκινεί τις διαδικασίες για handover. Όπως καταλαβαίνουμε όμως, θα έπρεπε να αλλάξει δίκτυο ο mobile node για να έχει πρόσβαση σε δίκτυο με μεγαλύτερο bandwidth και πιθανόν οικονομικότερο. Άρα, γενικεύοντας, το πότε θα γίνει το handover δεν έχει να κάνει μόνο με την διατήρηση της σύνδεσης, αλλά και με κριτήρια όπως κόστος και bandwidth.

Όπως αναφέραμε και παραπάνω, οι handover initiation αλγόριθμοι είναι υπεύθυνοι για την εκκίνηση της διαδικασίας handover, και οι handover execution αλγόριθμοι, περιγράφουν τον τρόπο με τον οποίο θα γίνει η διαδικασία του handover. Παρακάτω γίνεται ανάλυση των αλγορίθμων αυτών.

3.3 Αλγόριθμοι για Handover Initiation

Γενικότερα, υπάρχουν δύο είδη αλγόριθμων για handover initiation. Οι λεγόμενοι

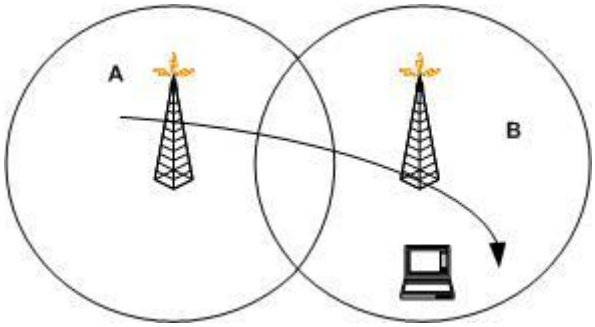
- Reactive : Όπου το handover εκτελείται αφού χαθεί η επαφή με το τρέχων δίκτυο, και το

- Proactive: Όπου το handover εκτελείται πριν χαθεί η επικοινωνία με το τρέχων δίκτυο.

Στις προδιαγραφές του Mobile IP περιγράφονται δύο handover initiation αλγόριθμοι. Ο Lazy Cell Switching και ο Eager Cell Switching οι οποίοι στηρίζονται στην περιοδική εκπομπή Agent Advertisements μηνυμάτων από τους foreign agents .

3.3.1 Lazy Cell Switching

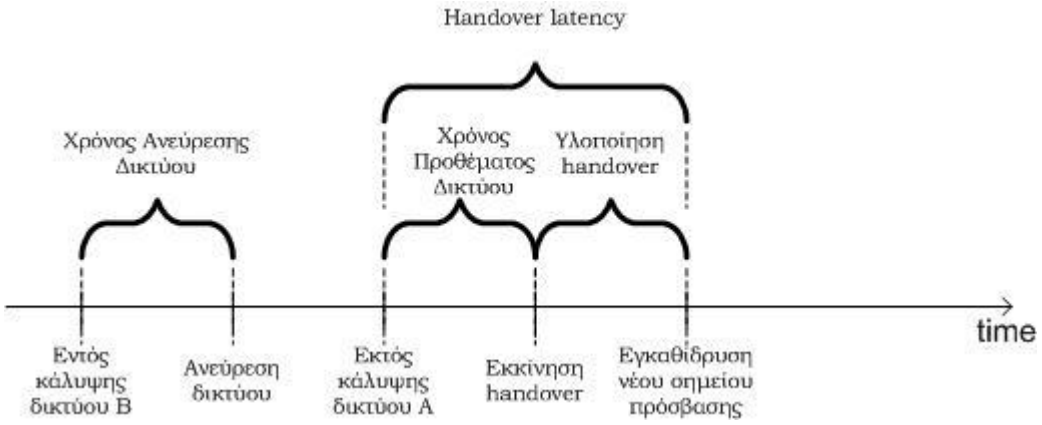
Ο Lazy Cell Switching αλγόριθμος, είναι ένας reactive αλγόριθμος. Σύμφωνα με τον αλγόριθμο αυτό, η διαδικασία του handover θα ξεκινήσει τη στιγμή που ο mobile node αντιληφθεί ότι δεν είναι εφικτή η σύνδεση με τον foreign agent στον οποίο ήταν συνδεδεμένος. Αυτό επιτυγχάνεται με την παρακολούθηση των agent advertisement μηνυμάτων που αποστέλλει ο foreign agent. Μόλις δηλαδή σταματήσει να λαμβάνει τα advertisement μηνύματα ο mobile node, αντιλαμβάνεται ότι έχει μετακινηθεί από την περιοχή κάλυψης του παλιού foreign agent και εκκινεί τη διαδικασία του handover.



Σχήμα 15

Στο σχήμα 15, απεικονίζεται ένα απλό σενάριο κίνησης. Ο mobile node μετακινείται από την κυψέλη A στην κυψέλη B. Όπως βλέπουμε υπάρχει αλληλοεπικάλυψη μεταξύ των κυψελών. Με τη βοήθεια του σχήματος 15, θα περιγράψουμε την διαδικασία του handover κάνοντας χρήση του Lazy Cell Switching

αλγόριθμου.



Σχήμα 16

Στο παραπάνω σχήμα, περιγράφεται η διαδοχή των γεγονότων που λαμβάνουν χώρα στο εν λόγω σενάριο κίνησης. Όπως μετακινείται ο mobile node, και είναι καταχωρημένος στο δίκτυο Α, εισέρχεται στην περιοχή αλληλοεπικάλυψης. Μετά από ένα χρονικό διάστημα «*Χρόνος Ανεύρεσης Δικτύου*», δέχεται agent advertisement μήνυμα από τον νέο foreign agent. Το χρονικό αυτό διάστημα, εξαρτάται άμεσα από την περίοδο εκπομπής των advertisement μηνυμάτων από τον agent. Δηλαδή, όσο πιο συχνά εκπέμπει advertisement μηνύματα ο agent, τόσο πιο σύντομα θα ανακαλύψει το νέο δίκτυο ο mobile node.

Συνεχίζοντας την πορεία του ο mobile node προς το δίκτυο Β, φεύγει από τα όρια της κυψέλης Α. Δεν γίνεται όμως άμεσα αντιληπτό από τον mobile node αλλά μετά από χρονικό διάστημα « *Χρόνος Προθέματος Δικτύου* ». Το χρονικό διάστημα αυτό, βασίζεται άμεσα στη διάρκεια ζωής (time to live) του προθέματος δικτύου. Κάθε φορά που ο mobile node λαμβάνει agent advertisement μήνυμα, στο μήνυμα αυτό αναφέρεται και ο χρόνος για τον οποίο θα είναι έγκυρη η Care-of-address που του έχει κατοχυρωθεί. Στη συνέχεια, αφού ο mobile node αντιληφθεί ότι δεν υπάρχει πρόσβαση πλέον μέσω του παλαιού foreign agent, εκκινεί τις διαδικασίες για το handover. Αφού ολοκληρωθεί το handover, αποκτά ο mobile node, μια νέα Care-of-Address από τον νέο foreign agent, και έτσι έχει αποκτήσει εκ νέου πρόσβαση στο Διαδίκτυο. Ως handover

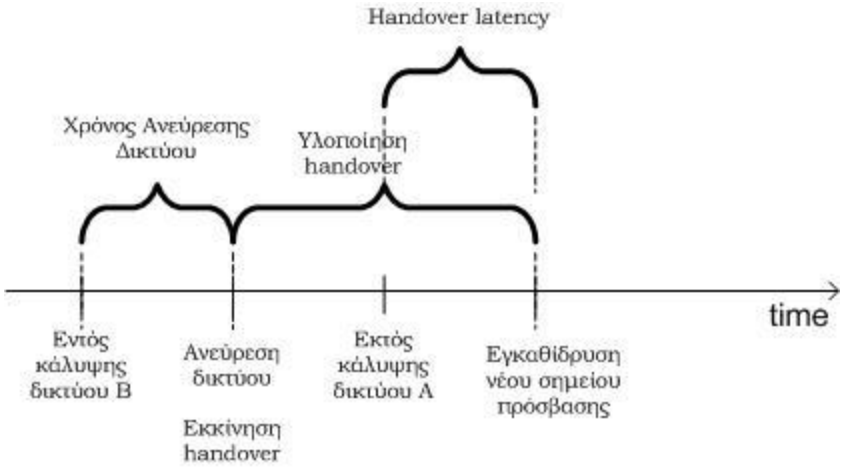
latency ορίζουμε το χρονικό διάστημα από τη στιγμή « *Εκτός κάλυψης δικτύου Α* », έως τη στιγμή « *Εγκαθίδρυση νέου σημείου πρόσβασης* », διότι στο χρονικό αυτό διάστημα, ο mobile node δεν έχει σημείο πρόσβασης και άρα δεν μπορεί ούτε να στείλει αλλά ούτε και να δεχτεί πακέτα.

Εάν παρατηρήσουμε το σχήμα 16, βλέπουμε ότι το handover latency εξαρτάται άμεσα από την time-to-live τιμή της Care-of-Address του δικτύου Α. Άρα, μπορούμε να υποθέσουμε ότι μία μικρή TTL τιμή θα ελάττωνε το handover latency. Αυτό όμως θα είχε ως αποτέλεσμα, ο foreign agent να αναγκαστεί να εκπέμπει συχνότερα advertisement μηνύματα, για να ανανεώνει την TTL. Οπότε, θα φορτώναμε την επικοινωνία mobile node με foreign agent, με μεγαλύτερη κίνηση σηματοδοσίας. Επιπλέον, στην πιθανή περίπτωση όπου ο mobile node, δεν λάβει έγκαιρα advertisement μήνυμα λόγω προβληματική ραδιοκάλυψης τη δεδομένη στιγμή, θα προβεί στη διαδικασία του handover, όντας όμως στην περιοχή κάλυψης του παλιού foreign agent. Άρα δηλαδή θα έχουμε προσπάθεια για handover τη στιγμή που αυτό δεν είναι αναγκαίο.

Συνοψίζοντας, βλέπουμε ότι κάνοντας χρήση του Lazy Cell Switching αλγόριθμου, πρέπει να βρεθεί μία χρυσή τομή στη TTL της Care-of-Address, έτσι ώστε να κρατείται σε χαμηλό επίπεδο το handover latency αλλά και να μην απαιτείται μεγάλη κίνηση σηματοδοσίας.

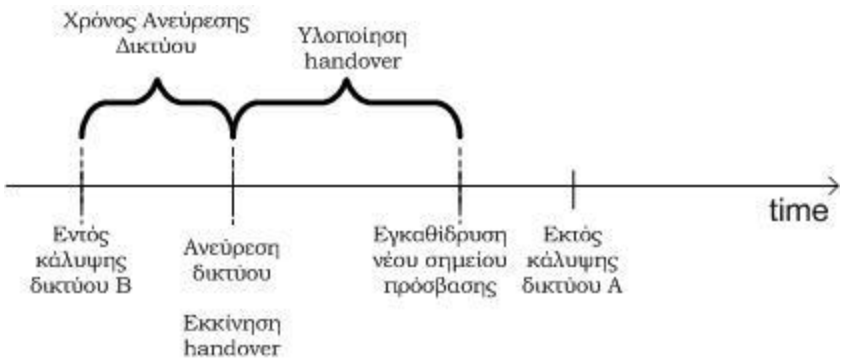
3.3.2 Eager Cell Switching

Ο άλλος handover initiation αλγόριθμος που περιγράφεται στις προδιαγραφές του Mobile IP, είναι ο Eager Cell Switching αλγόριθμος. Ο αλγόριθμος αυτός είναι ένας proactive αλγόριθμος. Δηλαδή, εκκινεί τη διαδικασία του handover, πριν χαθεί η επικοινωνία με το τρέχων δίκτυο. Για την περιγραφή του, ανατρέχουμε στο παράδειγμα κίνησης του σχήματος 15. Εδώ ο mobile node κινείται από την κυψέλη Α προς την κυψέλη Β, αλλά αυτή τη φορά κάνει χρήση του Eager Cell Switching αλγόριθμου.



Σχήμα 17

Με τη βοήθεια του σχήματος 17, θα αναλύσουμε τον αλγόριθμο. Μόλις ο mobile node εισέρχεται στην περιοχή αλληλοεπικάλυψης των κυψελών βρίσκεται και στην περιοχή κάλυψης της κυψέλης Β. Μετά από χρονικό διάστημα « *Χρόνος Ανεύρεσης Δικτύου* », λαμβάνει το πρώτο advertisement μήνυμα από τον νέο agent. Εκείνη τη στιγμή ο αλγόριθμος δίνει εντολή να εκκινήσει η διαδικασία του handover. Να σημειώσουμε εδώ, ότι mobile node, έχει ακόμη πρόσβαση μέσω της κυψέλης Α. Από τη στιγμή που βρίσκεται εκτός ορίων της κυψέλης Α και μετά από το χρονικό διάστημα «*Handover latency*», ο mobile node εγκαθιδρύει το νέο σημείο πρόσβασης με την κυψέλη Α, και είναι σε θέση να στείλει αλλά και να λάβει πακέτα. Όπως καταλαβαίνει κανείς και από το σχήμα 17, ο Eager Cell Switching αλγόριθμος είναι αποδοτικότερος από τον Lazy Cell Switching, διότι μειώνει κατά πολύ το handover latency, αφού οι διαδικασίες για το handover ξεκινούν νωρίτερα άρα και το handover ολοκληρώνεται νωρίτερα.



Σχήμα 18

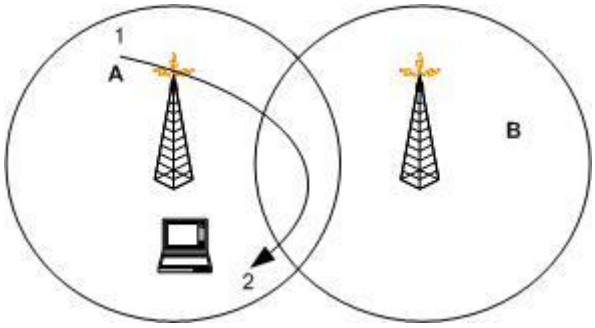
Στο σχήμα 17, υποθέσαμε ότι πρώτα χάνεται η επαφή με την κυψέλη Α και κατόπιν ολοκληρώνεται το handover δηλαδή αποκτά νέο σημείο πρόσβασης ο mobile node. Στο σχήμα 18, υποθέτουμε το σενάριο ότι πρώτα εγκαθιδρύεται το σημείο πρόσβασης με το δίκτυο Β και έπειτα χάνεται η κάλυψη από το δίκτυο Α. Αυτό μπορεί να συμβεί στην περίπτωση που ο mobile node κινείται με χαμηλή ταχύτητα ή η περιοχή αλληλοεπικάλυψης των δύο κυψελών είναι αρκετά μεγάλη οπότε προλαβαίνει να ολοκληρωθεί το handover. Όπως εύκολα παρατηρεί ο αναγνώστης, δεν υπάρχει καθόλου handover latency. Αν θέλουμε να είμαστε πιο αυστηροί, θα αναφερόμασταν σε μηδενικό handover latency, στην περίπτωση όπου συμπίπτουν χρονικά τα σημεία «*Εγκαθίδρυση νέου σημείου πρόσβασης*» και «*Εκτός κάλυψης δικτύου Α*» , ενώ στο συγκεκριμένο σενάριο, έχουμε αρνητικό handover latency, δηλαδή πρακτικά μηδενικό.

Εάν μάλιστα, υποθέσουμε την περίπτωση ότι ο mobile node λάβει advertisement μήνυμα από τον νέο foreign agent αμέσως μόλις εισέλθει στην περιοχή κάλυψης της κυψέλης Β, τότε το handover ολοκληρώνεται πολύ νωρίτερα.

Εάν δηλαδή, κρατήσουμε το χρόνο ολοκλήρωσης του handover σταθερό, αφού για αυτό δεν μπορούν να κάνουν τίποτα οι handover initiation αλγόριθμοι, παρατηρούμε ότι το handover latency μπορεί να ελαττωθεί ή και ακόμη να εκμηδενιστεί, είτε με αύξηση της περιοχής αλληλοεπικάλυψης, δηλαδή να τοποθετηθούν οι κυψέλες πιο κοντά η μία στην άλλη, είτε με συχνότερη εκπομπή advertisement μηνυμάτων από τους agent. Στην πρώτη περίπτωση, πυκνότερη γεωγραφική κατανομή των κυψελών, απαιτεί μεγάλο αριθμό κυψελών, το οποίο είναι

τουλάχιστον αντιοικονομικό. Στη δεύτερη περίπτωση, όπως αναφέραμε και παραπάνω, συχνή εκπομπή advertisement μηνυμάτων, οδηγεί σε αυξημένη κίνηση σηματοδοσίας. Άρα όπως φαίνεται και εδώ, πρέπει να βρεθεί η χρυσή τομή όσον αφορά στην κατανομή των κυψελών αλλά και στη συχνότητα εκπομπής μηνυμάτων ελέγχου.

Συνοψίζοντας, παραπάνω παρουσιάσαμε τους δύο handover initiation αλγόριθμους που περιγράφουν οι προδιαγραφές του Mobile IP. Από την ανάλυσή τους, εύκολα ο αναγνώστης καταλαβαίνει ότι ο Lazy Cell Switching υστερεί σε σχέση με τον Eager Cell Switching, αφού ο τελευταίος επιτυγχάνει μικρό handover latency και σε ειδικές περιπτώσεις μηδενικό. Ας δούμε όμως το σενάριο κίνησης που απεικονίζεται στο σχήμα 19.



Σχήμα 19

Στο παραπάνω σχήμα λοιπόν, ο mobile node βρίσκεται στο σημείο 1, και κινείται προς το σημείο 2. Όπως παρατηρούμε, το αρχικό σημείο και το τελικό σημείο βρίσκονται αποκλειστικά στην περιοχή κάλυψης της κυψέλης Α. Κατά την διάρκεια της διαδρομής, ο mobile node εισέρχεται στην περιοχή αλληλοεπικάλυψης. Δηλαδή καθ' όλη τη διάρκεια της διαδρομής, ο mobile node βρίσκεται εντός κάλυψης της κυψέλης Α και για κάποιο χρονικό διάστημα βρίσκεται εντός της περιοχής κάλυψης των κυψελών Α και Β.

Εάν ο initiation αλγόριθμος είναι ο Lazy Cell Switching, δεν γίνεται καμία διαδικασία για handover. Και αυτό διότι, όπως περιγράψαμε και παραπάνω, σύμφωνα με τον εν λόγω αλγόριθμο γίνεται handover μόνο στην περίπτωση που χαθεί η σύνδεση

με το δίκτυο, κάτι το οποίο δεν συμβαίνει στο συγκεκριμένο σενάριο οπότε και δεν χρειάζεται να γίνει handover.

Στην περίπτωση όπου γίνεται χρήση του Eager Cell Switching, εκκινούνται οι διαδικασίες για handover. Μόλις εισέλθει ο mobile node στην περιοχή αλληλοεπικάλυψης, και λάβει advertisement μήνυμα από το σταθμό Β, εκκινεί τη διαδικασία του handover. Στη συνέχεια, αφού πλέον είναι καταχωρημένος στη κυψέλη Β και βρίσκεται στην περιοχή αλληλοεπικάλυψης θα δεχτεί ένα advertisement μήνυμα από τον σταθμό Α. Οπότε, θα προβεί σε νέο handover και αφού ολοκληρωθεί θα καταχωρηθεί στον σταθμό Α. Καταλαβαίνουμε δηλαδή ότι όσο χρόνο ο mobile node βρίσκεται στη περιοχή αλληλοεπικάλυψης μεταξύ των κυψελών και άρα δέχεται advertisement μηνύματα και από τους δύο agent θα έχουμε απανωτά handover.Αυτό το φαινόμενο αναφέρεται και ως *ping pong effect*. Στη συνέχεια ο mobile node αφού φύγει από την περιοχή αλληλοεπικάλυψης, πιθανών να έχουμε ένα ακόμη handover ώστε να καταχωρηθεί τελικά στον σταθμό βάσης Α.

Από το παραπάνω σενάριο, καταλαβαίνουμε ότι δεν μπορούμε να αναδείξουμε ως βέλτιστο initiation αλγόριθμο ούτε τον Lazy αλλά ούτε και τον Eager Cell Switching, αν δεν γνωρίζουμε εκ των προτέρων το σενάριο κίνησης του mobile node, πράγμα αδύνατο. Δηλαδή, στις προδιαγραφές του Mobile IP δεν περιγράφεται ένας handover initiation αλγόριθμος, που να αντιμετωπίζει αποδοτικά όλα τα σενάρια κίνησης. Ο Lazy Cell Switching εμφανίζει μεγάλο handover latency αλλά δεν έχουμε ανεπιθύμητα handover , ενώ με τον Eager Cell Switching έχουμε μικρότερο ως και μηδενικό latency αλλά μπορεί να έχουμε πολλά ανεπιθύμητα handover. Αναδεικνύεται δηλαδή η ανάγκη εύρεσης ενός πιο προηγμένου initiation αλγόριθμου, που να αντιμετωπίζει όλα τα πιθανά σενάρια κίνησης όσο το δυνατόν καλύτερα, δηλαδή να ελαχιστοποιεί τα ανεπιθύμητα handover αλλά και να διατηρεί σε μικρές τιμές το handover latency.

Για να επιτύχουμε όμως τα παραπάνω, οι πληροφορίες από το επίπεδο δικτύου (Layer 3), δηλαδή τα agent advertisement μηνύματα, δεν επαρκούν. Χρειαζόμαστε πληροφορίες που να δίνουν μία εικόνα της ζεύξης μεταξύ mobile node και foreign

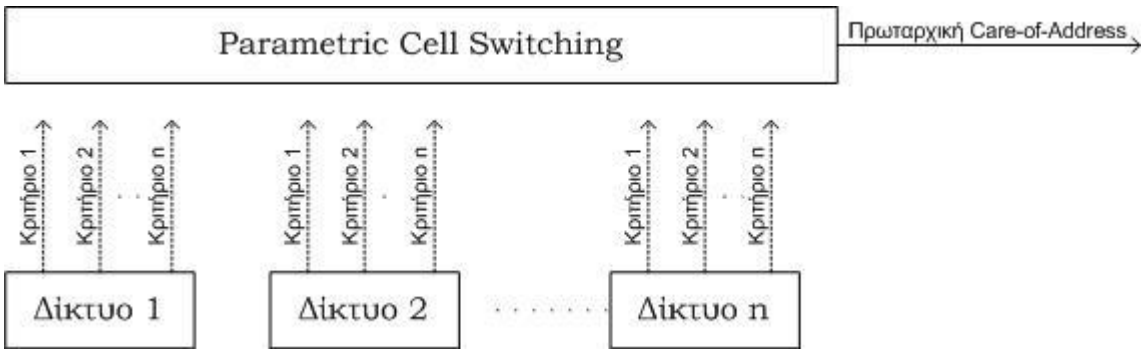
agent και κατά μία έννοια να προβλέπουν την κίνηση του mobile node, έτσι ώστε να προβαίνουν στο initiation νωρίτερα και αποδοτικότερα.

Ένας τέτοιος handover initiation αλγόριθμος, πρέπει να λαμβάνει υπόψιν του δεδομένα και από το επίπεδο δικτύου αλλά και από το επίπεδο σύνδεσης (Layer 2), κάτι το οποίο υποστηρίζει και ο Charles E. Perkins, ένας από τους πρώτους σχεδιαστές του Mobile IP. Ένας τέτοιος αλγόριθμος, έχει προταθεί από τους Torben Wittrup Andersen και Anders Lildballe στη μεταπτυχιακή τους διατριβή " Seamless Handoff in Mobile IPv6 " [17]. Ο αλγόριθμος που προτείνουν ονομάζεται « Parametric Cell Switching ».

3.3.3 Parametric Cell Switching

Οι πληροφορίες που μπορεί να μας παρέχει το επίπεδο δικτύου, όπως η ισχύς του σήματος ή το bit error rate, μπορούν να είναι διαθέσιμες συνέχεια και μπορούν να μετρηθούν σε οποιαδήποτε συχνότητα, παρέχοντάς μας έτσι μία μετρήσιμη πληροφορία για την ποιότητα της φυσικής σύνδεσης. Μία τέτοια πληροφορία, επιτρέπει στον mobile node να ανιχνεύσει γρήγορα την εξασθένηση της ζεύξης και να προβεί σε handover προτού χαθεί η πρόσβαση με το δίκτυο.

Με τη βοήθεια του σχήματος 3-8, θα περιγράψουμε τη βασική αρχή λειτουργίας του Parametric Cell Switching αλγόριθμου.



Σχήμα 20

Η κύρια δομή του αλγόριθμου, παρουσιάζεται στο παραπάνω σχήμα. Αποτελείται από τον Parametric Cell Switching αλγόριθμο, από διάφορα διαθέσιμα δίκτυα και από κάποια κριτήρια για κάθε δίκτυο, τα οποία λαμβάνει υπόψιν του ο αλγόριθμος.

Η κύρια ευθύνη του αλγόριθμου, είναι να επιλέξει την care-of-address του «καλύτερου» δικτύου ως πρωταρχική care-of-address. Για κάθε δίκτυο, ο αλγόριθμος κάνει μια αποτίμηση των κριτηρίων και υπολογίζει ποιο είναι το «καλύτερο» δίκτυο. Εάν η care-of-address του καλύτερου δικτύου, δεν είναι η πρωταρχική care-of-address, τότε εκκινούνται οι διαδικασίες για handover για την απόκτηση αυτής. Στην αντίθετη περίπτωση, δεν έχουμε handover.

Ο αντικειμενικός σκοπός αυτού του proactive initiation αλγόριθμου, είναι να εκκινεί τις διαδικασίες για handover, προτού χαθεί η σύνδεση με το τρέχων δίκτυο και ως εκ τούτου να μην έχουμε απώλεια πακέτων αλλά και να διατηρεί το πλήθος των handover σε λογικό επίπεδο.

Επιλογή των κριτηρίων.

Ως κριτήριο, εννοούμε την πληροφορία που παρέχεται στον Parametric Cell Switching αλγόριθμο, η οποία περιγράφει ένα συγκεκριμένο χαρακτηριστικό του δικτύου. Τα κριτήρια τα οποία έχουν επιλεγεί για το Parametric Cell Switching είναι:

- Ο λόγος σήματος προς θόρυβο ή SNR. (Link Layer)
- Round trip time από τον mobile node προς τον agent. (Network Layer)
- Η διάρκεια της διαθεσιμότητας του δικτύου.(Network Layer)
- Το κόστος χρήσης του δικτύου. (Network Layer)

Όπως παρατηρούμε, τρία κριτήρια βρίσκονται στο επίπεδο δικτύου και ένα στο επίπεδο σύνδεσης. Παρακάτω αναλύουμε το κάθε κριτήριο και τι πληροφορίες μπορούμε να πάρουμε από το καθένα.

Λόγος Σήματος προς Θόρυβο

Ο λόγος σήματος προς θόρυβο, είναι ένα μέγεθος το οποίο μας δείχνει το πόσο ισχυρό είναι ένα σήμα σε σχέση με το θόρυβο. Όπως είναι λογικό, για να μετρηθεί ένα τέτοιο μέγεθος, απαιτείται η μέτρηση της ισχύος του θορύβου και ξεχωριστά η μέτρηση της ισχύος του σήματος. Εάν σαν κριτήριο είχαμε μόνο την ισχύ του σήματος, τότε σε ένα περιβάλλον με έντονο ηλεκτρομαγνητικό θόρυβο πιθανών να μετρούσαμε μεγάλη ισχύ σήματος και να είχαμε την λανθασμένη εντύπωση, ότι η ποιότητα της ζεύξης είναι καλή. Να σημειώσουμε εδώ ότι οι σχεδιαστές του Parametric Cell Switching αλγόριθμου, έχουν λάβει υπόψιν τους μόνο το downlink SNR.

Round Trip Time

Με τον όρο round trip time, εννοούμε τον χρόνο που διανύει ένα πακέτο από τον mobile node προς τον agent και από τον agent στον mobile node. Το παραπάνω μέγεθος μπορεί να μετρηθεί πολύ απλά με ένα ping (echo request) από τον mobile node προς τον agent. Ένα επιτυχημένο ping δείχνει ότι υπάρχει επικοινωνία μεταξύ των δύο. Ο χρόνος αυτός επηρεάζεται άμεσα από το εύρος ζώνης του δικτύου αλλά και από την κίνηση στο δίκτυο. Δηλαδή με το round trip time ο initiation αλγόριθμος μπορεί να κάνει μια εκτίμηση των δύο παραπάνω και έτσι μπορεί απλά να αποφασίσει ποιο από τα διαθέσιμα δίκτυα προσφέρει το καλύτερο bandwidth. Να σημειώσουμε εδώ, ότι πολλοί routers, για να αποφύγουν επιθέσεις τύπου Denial of Service, έχουν ρυθμιστεί έτσι ώστε να μην ανταποκρίνονται σε echo request.

Διάρκεια της διαθεσιμότητας του δικτύου.

Η διάρκεια της διαθεσιμότητας του δικτύου είναι το χρονικό διάστημα στο οποίο ο mobile node έχει πρόσβαση στο δίκτυο αυτό. Δηλαδή, ο αλγόριθμος κρατάει

σε μία μνήμη το χρονικό διάστημα για κάθε agent ξεχωριστά, στον οποίο βρισκόταν κατά το παρελθόν ο mobile node. Ο αλγόριθμος, μπορεί να επιλέξει δηλαδή το δίκτυο όπου στο παρελθόν έχει βρεθεί για μεγάλο χρονικό διάστημα. Όπως καταλαβαίνουμε, αυτό το κριτήριο έχει νόημα μόνο για δίκτυα τα οποία ξανά επισκέπτεται ο mobile node και βρίσκει σπουδαία εφαρμογή όταν ο mobile node ακολουθεί ένα συγκεκριμένο σενάριο κίνησης. Προχωρώντας παραπέρα, θα μπορούσαμε να πούμε ότι με το κριτήριο αυτό *εκπαιδεύεται* ο Parametric Cell Switching αλγόριθμος.

Κόστος χρήσης του δικτύου

Με τον όρο αυτό, όπως εύκολα καταλαβαίνουμε, εννοούμε το κόστος χρήσης του δικτύου είτε σε ευρώ ή δολάρια ανά Megabyte που ο mobile node δέχεται ή λαμβάνει είτε σε ευρώ ή δολάρια ανά λεπτό χρήσης του δικτύου. Εμπεριέχοντας αυτό το κριτήριο, ο αλγόριθμος λαμβάνει υπόψιν και το κριτήριο του κόστους.

Διαβάθμιση των κριτηρίων

Παραπάνω, παραθέσαμε όλα τα κριτήρια τα οποία λαμβάνει υπόψιν του ο Parametric Cell Switching αλγόριθμος για να επιλέξει το κατάλληλο δίκτυο και να εκκινήσει, αν χρειάζεται, τη διαδικασία του handover. Ο αλγόριθμος όμως δεν δίνει την ίδια βαρύτητα σε όλα τα κριτήρια. Σύμφωνα με τον αλγόριθμο αυτό, κάθε ένα handover κριτήριο αφού μετρηθεί, σταθμίζεται σύμφωνα με το ειδικό βάρος που δίνεται σε κάθε κριτήριο και κατόπιν όλα μαζί αθροίζονται για να δώσουν τη βαθμολογία για κάθε δίκτυο. Εάν η Care-of-Address του δικτύου με τη μεγαλύτερη βαθμολογία δεν είναι η πρωταρχική, τότε έχουμε handover για την απόκτηση αυτής της Care-of-address.

Εάν οι βαθμολογίες δύο ή και περισσότερων δικτύων, αυξομειώνονται στην ίδια περίπου τιμή, τότε είναι πιθανό σε κάθε υπολογισμό των handover κριτηρίων, να έχουμε handover, να εμφανίζεται δηλαδή το ping pong effect. Για την αποφυγή τέτοιων

φαινομένων, οι σχεδιαστές του Parametric Cell Switching αλγόριθμου, έχουν θεσπίσει ως όριο για την εκκίνηση handover, το καλύτερο δίκτυο να έχει βαθμολογία τουλάχιστον 10% της κλίμακας από το τρέχων δίκτυο. Δηλαδή σε μία κλίμακα [-10,10] το καλύτερο δίκτυο πρέπει να είναι τουλάχιστον δύο βαθμούς καλύτερο από το τρέχων δίκτυο για να γίνει handover.

Η γενική μαθηματική έκφραση για την αποτίμηση της βαθμολογίας του κάθε δικτύου είναι:

$$\mathrm{S} = \sum_{\mathrm{i}=1}^{m} \mathrm{W_i} \cdot P_i$$

όπου S είναι η βαθμολογία του κάθε δικτύου, P_i είναι η συνεισφορά του i-οστού handover κριτηρίου και W_i είναι το ειδικό βάρος του i-οστού κριτηρίου. Για τα συγκεκριμένα τέσσερα handover κριτήρια που λαμβάνει υπόψιν του ο Parametric Cell Switching αλγόριθμος η μαθηματική έκφραση είναι:

$$S = W_{SNR}P_{SNR} + W_{RTT}P_{RTT} + W_{Avail}P_{Avail} + W_{\mathrm{P}rice}P_{\mathrm{P}rice}$$

όπου SNR δηλώνει λόγος σήματος προς θόρυβο, RTT δηλώνει τον round trip time, Avail δηλώνει τη διάρκεια της διαθεσιμότητας του δικτύου και Price το κόστος του δικτύου.

3.3.4 Συμπεράσματα

Συνοψίζοντας ως εδώ, έχουμε παρουσιάζει τους δύο handover initiation αλγόριθμους που περιγράφονται στις προδιαγραφές του Mobile IP, τον Lazy Cell Switching και τον Eager Cell Switching. Είδαμε τα πλεονεκτήματα και τις αδυναμίες των δύο αυτών αλγορίθμων, αλλά η ποιο σημαντική παρατήρηση είναι ότι, δεν μπορεί να υπάρξει ένα initiation αλγόριθμος που να αντιμετωπίζει αποδοτικά όλα τα πιθανά σενάρια κίνησης βασιζόμενος μόνο στα agent advertisement μηνύματα που λαμβάνει. Υπήρξε η ανάγκη να ληφθούν υπόψιν και άλλοι παράγοντες και κυρίως δεδομένα από

το επίπεδο σύνδεσης δεδομένων. Παρουσιάσαμε έναν τέτοιο αλγόριθμο, τον Parametric Cell Switching, ο οποίος δεν βασίζεται σε advertisement μηνύματα, αλλά σε άλλες πληροφορίες τις οποίες έχουμε παρουσιάσει αναλυτικά παραπάνω. Με άλλα λόγια, ο αλγόριθμος αυτός εφαρμόζει την cross-layer σχεδίαση, δηλαδή συλλέγει δεδομένα από δύο επίπεδα. Το επίπεδο δικτύου (Network Layer) και το επίπεδο σύνδεσης (Link Layer). Επίσης, ο αλγόριθμος αυτός παρουσιάζει μια ευελιξία η οποία οφείλεται στα ειδικά βάρη των κριτηρίων. Μεταβάλλοντας τα βάρη αυτά, μπορούμε στην ουσία να αλλάξουμε τον αλγόριθμο.

3.4 Handover Execution

Μέχρι τώρα, έχουμε δει τους handover initiation αλγόριθμους. Δηλαδή το πότε να γίνει το handover και το εάν πρέπει να γίνει. Δεν έχουμε δει το πώς γίνεται το handover. Εύκολα ο αναγνώστης αντιλαμβάνεται, ότι ο τρόπος που θα γίνει το handover έχει πολύ περισσότερο αντίκτυπο στο handover latency και στο packet loss, από το πότε θα γίνει το handover.

Ας δούμε, με βάση αυτά που έχουμε πει μέχρι τώρα, το τρόπο με τον οποίο γίνεται το handover. Μόλις ο initiation αλγόριθμος ξεκινήσει τη διαδικασία του handover, ο mobile node στέλνει ένα Registration Request μήνυμα στον Home agent, με την νέα πλέον Care-of-Address. Στη συνέχεια, ο Home agent θα καταχωρήσει τον mobile node με την νέα του διεύθυνση και θα αποστείλει με τη σειρά του ένα Registration Reply μήνυμα προς τον mobile node, ενημερώνοντας τον ότι πλέον μπορεί να κάνει χρήση της νέας του Care-of-Address. Όπως είναι λογικό, ο home agent θα στέλνει πλέον τα πακέτα που προορίζονται για τον mobile node στη νέα του Care-of-Address χρησιμοποιώντας τεχνικές tunneling, όπως περιγράψαμε παραπάνω.

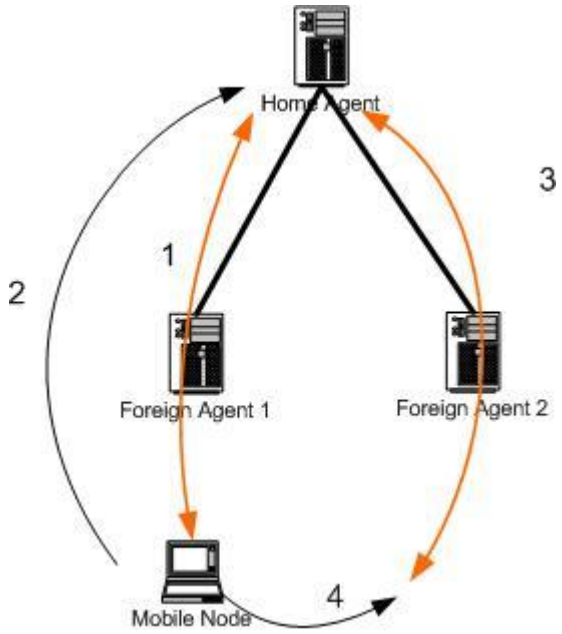
Καταλαβαίνουμε δηλαδή, ότι μεγάλο μέρος του handover latency, οφείλεται στη διαδικασία του νέου registration. Επίσης, πακέτα τα οποία έχουν αποσταλεί προς την παλιά Care-of-Address, θα χαθούν. Άρα, με μία πρώτη προσέγγιση στο πρόβλημα, μπορούμε να υποθέσουμε ότι εάν θέλουμε να ελαττώσουμε το handover latency,

πρωταρχικός στόχος είναι να ελαττώσουμε την διάρκεια του registration.

Παρακάτω, παρουσιάζουμε και αναλύουμε τις διάφορες προτάσεις που έχουν γίνει για τη λύση του παραπάνω προβλήματος.

3.4.1 Fast Handover via Simultaneous Bindings

Η πρόταση αυτή [8], επιτρέπει σε έναν mobile mode, να διατηρεί πολλαπλά bindings ταυτόχρονα με τον home agent, για νέους foreign agent όταν αναμένεται να γίνει handover. Στο σχήμα 21, περιγράφεται η βασική ιδέα.



Σχήμα 21

Ο mobile node, ο οποίος είναι συνδεδεμένος στον foreign agent 1, ανακαλύπτει τον foreign agent 2. Οπότε στέλνει, ένα registration request μήνυμα προς τον home agent, με τη νέα Care-of-Address και τον ενημερώνει για simultaneous binding. Οπότε με τη σειρά του ο home agent στέλνει τα πακέτα που προορίζονται για τον mobile node και στις δύο κατευθύνσεις. Στη συνέχεια, ο mobile node θα διακόψει την επικοινωνία του με τον foreign agent 1 και θα μεταβεί στον foreign agent 2. Με τη σειρά του, ο

home agent μετά από κάποιο χρονικό διάστημα θα σταματήσει να στέλνει τα πακέτα στην παλιά Care-of-Address.

Η τεχνική αυτή λέγεται “make before break”, δηλαδή το handover ολοκληρώνεται προτού χαθεί η επικοινωνία του mobile node με τον παλιό foreign agent. Το handover latency, περιορίζεται μόνο στο χρονικό διάστημα της φυσικής αποσύνδεσης με το παλιό δίκτυο.

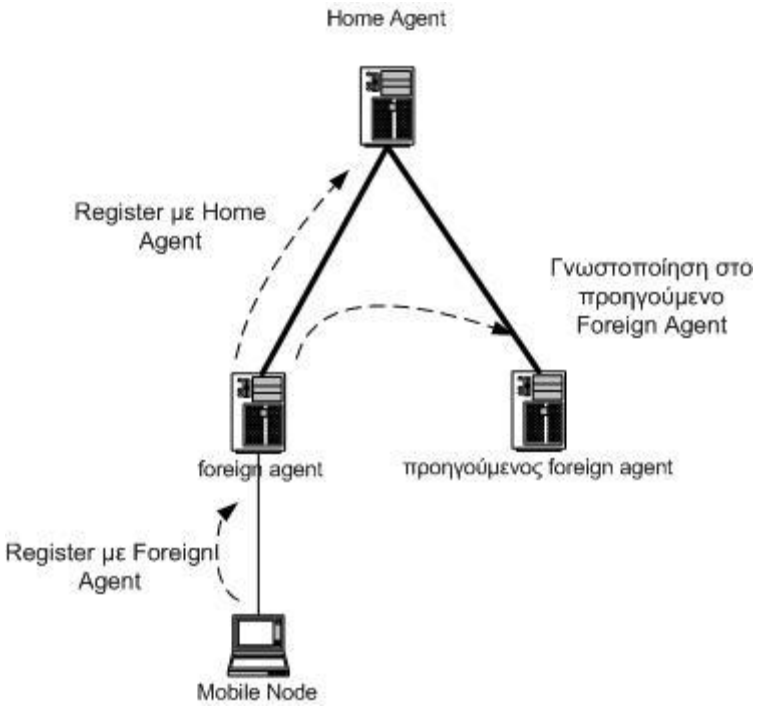
Το μειονέκτημα της προσέγγισης αυτής, είναι ότι μπορεί να βρει εφαρμογή μόνο στις περιπτώσεις όπου ο mobile node γνωρίζει “a priori” ότι πρόκειται να αλλάξει δίκτυο.

3.4.2 Foreign Agent Smooth Handover based on Route Optimization

Κατά τη διάρκεια του handover, τα πακέτα τα οποία προορίζονται για τον mobile node και στέλνονται στην παλιά Care-of-Address, θα χαθούν μέχρι να καταχωρηθεί η νέα Care-of-Address στον home agent.

Ακόμη και στην περίπτωση που εφαρμόζεται η route optimization τεχνική, τα πακέτα θα χαθούν διότι ο mobile node δεν μπορεί αμέσως να στείλει bindings update μηνύματα σε όλους τους correspondent nodes με τους οποίους επικοινωνεί για να τους ενημερώσει για την νέα του Care-of-Address.

Η πρόταση που παρουσιάζουμε εδώ [5], εμπλουτίζει θα λέγαμε την route optimization τεχνική, έτσι ώστε να έχουμε smooth handovers. Σύμφωνα με την τεχνική αυτή, οι foreign agent διατηρούν μία binding cache όπου εκεί αποθηκεύουν τα τρέχοντα bindings (τις νέες Care-of-Addresses) για τους mobile nodes με τους οποίους συνεργαζόντουσαν στο παρελθόν. Οι foreign agents συνεργάζονται μεταξύ τους έτσι ώστε να επιτευχθεί το smooth handover, προτού ακόμη η νέα Care-of-address καταχωρηθεί στον home agent.



Σχήμα 22

Με τη βοήθεια του σχήματος 22, περιγράφουμε παρακάτω την εν λόγω τεχνική. Ο mobile node, μόλις μεταβεί στον νέο foreign agent, του δίνει εντολή να ενημερώσει τον προηγούμενο foreign agent, με ένα μήνυμα του τύπου binding update, για τη νέα του Care-of-Address. Μόλις ενημερωθεί ο προηγούμενος agent, αμέσως δημιουργείται μία εγγραφή στην binding cache του. Οπότε, μόλις ο προηγούμενος foreign agent δεχθεί tunneled πακέτο που προορίζεται για τον εν λόγω mobile node, το αποθυλακώνει, και κάνοντας χρήση tunneling τεχνικών το στέλνει στον νέο foreign agent, δηλαδή στη νέα του Care-of-Address. Έτσι προλαμβάνεται η απώλεια πακέτων.

Στην περίπτωση που ο προηγούμενος foreign agent, λάβει πακέτο που έχει προορισμό τον mobile node, αλλά δεν υπάρχει καταχώρηση στην binding cache για τη νέα του διεύθυνση ή είναι εκπρόθεσμη, τότε ο foreign agent ξανά-ενθυλακώνει το πακέτο (με διεύθυνση πηγής πλέον την care-of-address του foreign agent) και το στέλνει στον home agent διαμέσου ενός ειδικού tunnel, υποδεικνύοντας την ειδική μεταχείριση που πρέπει να έχει το πακέτο από τον home agent. Εάν τώρα, η διεύθυνση πηγής, δηλαδή η care-of-address του foreign agent, ταιριάζει με το τελευταίο

registration binding του mobile node, ο home agent αντιλαμβάνεται ότι το registration binding που διατηρεί είναι απαρχαιωμένο και το πακέτο απορρίπτεται. Αρά σε αυτή την περίπτωση έχουμε απώλεια πακέτων.

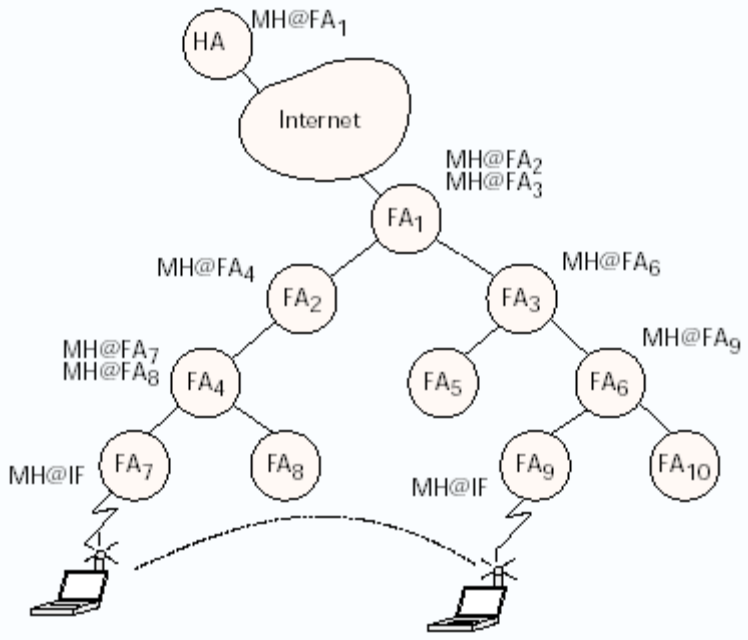
Εάν τώρα, η διεύθυνση πηγής του πακέτου δεν ταιριάζει με τη τελευταία registration binding, τότε ο home agent αντιλαμβάνεται ότι η binding cache του foreign agent είναι απαρχαιωμένη. Οπότε, ο home agent, μέσω tunnel, στέλνει το πακέτο στη σωστή Care-of-address, δηλαδή σε αυτήν που έχει καταχωρημένη στην binding cache του.

Το μειονέκτημα αυτής της μεθόδου είναι ότι, όταν ληφθούν πακέτα στον foreign agent μόλις ο mobile node έχει αποχωρήσει από αυτόν, και πριν ενημερωθεί με binding update μήνυμα, το αναγκαστικά τα πακέτα αυτά χάνονται.

Στη συνέχεια περιγράφουμε έναν μηχανισμό handover, ο οποίος είναι μία επέκταση θα λέγαμε του Foreign Agent Smooth Handover.

3.4.3 Optimised Smooth Handover based on Hierarchical Mobility Management

Το Hierarchical Mobility Management [10], εισάγει την ιεραρχική αρχιτεκτονική των foreign agents, η οποία είναι μία δενδροειδής δομή όπως φαίνεται και στο σχήμα 23. Επιτρέπει στον mobile node να διαπεραιώνει registrations τοπικά έτσι ώστε να μειωθεί η κίνηση μηνυμάτων ελέγχου μεταξύ αυτού και του home agent. Πολλές φορές στη βιβλιογραφία συναντάται και ως *Περιφερειακή Καταχώρηση (Regionalized Registration)*. Η δενδροειδής αυτή δομή, βελτιώνει κατά πολύ και την απόδοση του handover όπως θα δούμε και παρακάτω.



Σχήμα 23

Στο σχήμα 23, απεικονίζεται η ιεραρχική δομή των foreign agents. Τα πακέτα που προορίζονται για τον mobile node, στέλνονται από τον home agent (ή τον correspondent node αν έχουμε route optimization) μέσω tunnel προς τον Gateway Foreign Agent, ο οποίος στο σενάριο που εξετάζουμε είναι ο FA1. Στη συνέχεια, ο κάθε foreign agent αποθυλακώνει και ενθυλακώνει τα πακέτα για να τα στείλει στους foreign agents που βρίσκονται ένα επίπεδο κάτω από αυτόν. Αυτό γίνεται συνέχεια, μέχρι τα πακέτα να φθάσουν στον προορισμό τους, δηλαδή στον mobile node. Αυτή είναι η βασική αρχή λειτουργίας της αρχιτεκτονικής Hierarchical Foreign Agents. Ας δούμε τι γίνεται στην περίπτωση που ο mobile node αλλάζει δίκτυο. Όταν ο mobile node μετακινηθεί σε ένα νέο δίκτυο, τότε καταχωρεί τη νέα του Care-of-Address στον κοινό foreign agent μεταξύ παλαιού και νέου foreign agent. Δηλαδή, στο σενάριο του σχήματος 23, όπου ο mobile node κινείται από τον FA7 στον FA9, κοινός κόμβος είναι ο FA1. Εάν η μετακίνηση ήταν από τον FA7 στον FA8, τότε η καταχώρηση της νέας Care-of-Address, θα γινόταν στον FA4. Φαίνεται ξεκάθαρα δηλαδή, ότι η αρχιτεκτονική αυτή μειώνει κατά πολύ το χρόνο που απαιτείται για το registration σε σενάρια micromobility. Σε περιπτώσεις όμως, όπου ο mobile node κινείται από domain σε ένα

άλλο domain, η αρχιτεκτονική αυτή δεν είναι εφαρμόσιμη. Ένα τέτοιο σενάριο όμως, συμβαίνει σπανιότερα, οπότε το Mobile IP μπορεί να το διαχειριστεί χωρίς κανένα πρόβλημα.

Το optimized smooth handover επιτυγχάνεται, με την ενσωμάτωση της Foreign Agent Smooth Handover μεθόδου, που περιγράψαμε στο προηγούμενο εδάφιο, στην Hierarchical Foreign Agents αρχιτεκτονική. Επίσης, η τεχνική αυτή απαιτεί από τον προηγούμενο foreign agent τη δυνατότητα buffering.

Δηλαδή, όταν ο mobile node κάνει register την νέα του Care-of-Address, ζητά από τον νέο foreign agent να γνωστοποιήσει στον προηγούμενο foreign agent τη νέα του διεύθυνση. Σύμφωνα μεν όσα περιγράψαμε παραπάνω, ο νέος foreign agent θα στείλει ένα binding update μήνυμα στον παλιό foreign agent, ο οποίος με τη σειρά του θα στείλει ένα binding acknowledgement μήνυμα στον mobile node. Οπότε, όσα πακέτα σταλθούν στον παλιό foreign agent, εκείνος θα τα στείλει στον νέο foreign agent. Αλλά το πρόβλημα βρίσκεται στα πακέτα που λαμβάνει ο foreign agent μόλις έχει φύγει ο mobile node και πριν προλάβει ο πρώτος να ενημερωθεί για την νέα Care-of-address του τελευταίου. Για να λυθεί αυτό το πρόβλημα, πρέπει οι foreign agents να εφοδιασθούν με Forwarding Buffers. Ο foreign agent δηλαδή, θα διατηρεί ένα αντίγραφο των πακέτων που στέλνει στον mobile node, στον buffer. Οπότε, μόλις ο παλιός foreign agent λάβει binding update από τον νέο foreign agent, θα στείλει τα πακέτα που βρίσκονται στον buffer, όπως επίσης και τα πακέτα τα οποία λαμβάνει και προορίζονται για τον mobile node. Να σημειώσουμε εδώ, ότι ο mobile node ενημερώνει τον παλιό foreign agent για το τελευταίο πακέτο που έλαβε, έτσι ώστε να μην λάβει διπλά πακέτα.

Η προσέγγιση αυτή έχει το σημαντικό πλεονέκτημα ότι μειώνει των χρόνο του registration κατά πολύ όπως επίσης και την πιθανότητα να έχουμε χαμένα πακέτα. Όμως απαιτεί από τον foreign agent να διατηρεί forwarding buffer και είναι μεγάλη η πιθανότητα να έχουμε υπερχείλιση αυτού.

3.4.4 Position Leverage Smooth Handover Algorithm

Ο Position Leverage Smooth Handover αλγόριθμος [12], αντιμετωπίζει την κινητικότητα είτε μέσα στο ίδιο domain (intra-domain) είτε ανάμεσα σε διαφορετικά domain (inter-domain). Τα δύο αυτά σενάρια αντιμετωπίζονται ξεχωριστά. Ο αλγόριθμος αυτός, υποθέτει ότι τα δίκτυα είναι συγκροτημένα σε domains, όπου το καθένα έχει υιοθετήσει την hierarchical foreign agent αρχιτεκτονική. Επίσης, απαιτεί από κάθε router, συσκευές εντοπισμού για να εξασφαλίζεται το smooth handover. Ας δούμε όμως, τη λειτουργία του αλγόριθμου αυτού.

Στην κορυφή κάθε ιεραρχίας, βρίσκεται ο Domain Foreign Agent ο οποίος διατηρεί μία λίστα (visitor list) με τους mobiles nodes που βρίσκονται στο domain με τις Care-of- διευθύνσεις του καθενός. Ο agent αυτός λαμβάνει τα ενθυλακωμένα πακέτα από τον home agent, τα αποθυλακώνει, τα επαναενθυλακώνει, και τα αποστέλλει στον κατάλληλο foreign agent, βασισμένος στην visitor list. Επίσης, διατηρεί και τον Location-Foreign Agent πίνακα, όπου εκεί είναι καταχωρημένες οι διευθύνσεις των foreign agents αλλά και πληροφορίες για την τοποθεσία του καθένα. Ο κάθε foreign agent, που βρίσκεται ιεραρχικά χαμηλότερα, διατηρεί μία visitor list με τους mobile nodes που είναι συνδεδεμένοι με αυτόν. Κάθε foreign agent, στέλνει περιοδικά ένα σήμα (beacon)σε κάθε mobile node. Εάν δεν λάβει απάντηση από έναν mobile node που είναι καταχωρημένος στην visitor list, τότε η εγγραφή αυτή διαγράφεται από την λίστα και καταχωρείται σε μία cache. Η εγγραφή στη cache έχει περιστασιακή διάρκεια ζωής. Ο κάθε foreign agent διατηρεί επίσης και έναν Location-Foreign Agent πίνακα, παρόμοιο με αυτόν του Domain Foreign Agent.

Αφού αναφέραμε τις απαιτήσεις του Position Leverage Handover αλγόριθμου στην αρχιτεκτονική του δικτύου, ας δούμε πως βελτιστοποιεί το handover.

Όταν ο mobile node μετακινείται σε foreign agent του ίδιου domain, δηλαδή έχουμε intra-domain handover, καταχωρεί την νέα του Care-of-Address στον Domain Foreign Agent. Ο Domain Foreign Agent, είναι υπεύθυνος για την αποστολή των ενθυλακωμένων πακέτων προς τον mobile node. Ο Domain Foreign Agent,

συμβουλεύεται τον Location-Foreign Agents πίνακα, και έτσι αποστέλλει τα πακέτα προς τον foreign agent που «σερβίρει» τον mobile node, αλλά και στους διπλανούς στον mobile node, foreign agents.Οι γειτονικοί foreign agents με τη σειρά τους, κρατούν σε ένα buffer τα πακέτα αυτά και περιοδικά αδειάζουν το buffer. Με τον τρόπο αυτό, σε ένα handover θα είναι διαθέσιμα τα πακέτα από τον νέο foreign agent, και έτσι μειώνεται κατά πολύ η πιθανότητα για να έχουμε απώλεια πακέτων.

Ο αλγόριθμος αυτός δηλαδή, μειώνει στο ελάχιστο την κίνηση σηματοδοσίας από τον home agent στον mobile node, αν και αυξάνεται η κίνηση όμως μέσα στο domain.

Για την intra-domain κίνηση, δηλαδή όταν ο mobile node μετακινείται σε ένα νέο domain, η αντιμετώπιση είναι διαφορετική. Όταν, ο mobile node επισκεφθεί ένα νέο domain, στέλνει στον Domain Foreign Agent ένα registration request μήνυμα, συμπεριλαμβάνοντας μέσα την διεύθυνση του προηγούμενος Domain Foreign Agent αλλά και τη διεύθυνση του home agent. Ο νέος Domain Foreign Agent, κάνοντας χρήση του Global Tube αλγόριθμου, θα αποφασίσει αν θα στείλει το registration στον προηγούμενο Domain Foreign Agent ή στον home agent. Σύμφωνα με τον Global Tube αλγόριθμο, αν ο προηγούμενος Domain Foreign Agent βρίσκεται σχετικά κοντά με τον home agent, τότε το registration θα σταλεί στον home agent.

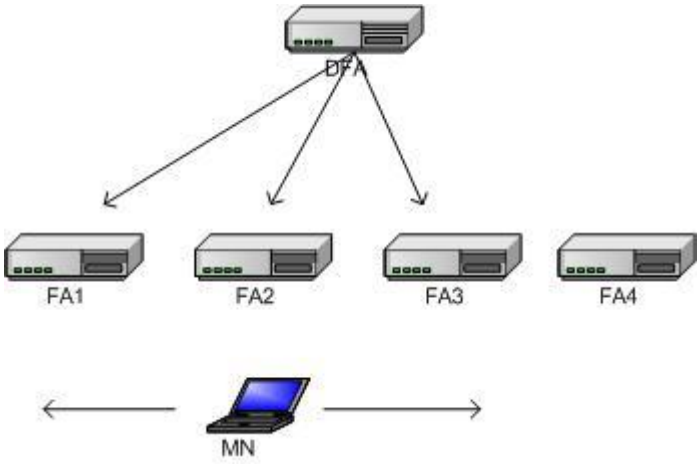
Στην αντίθετη περίπτωση, δηλαδή όταν ο προηγούμενος Domain Foreign Agent βρίσκεται σε μεγάλη απόσταση από τον home agent, τότε το registration θα σταλεί στον προηγούμενο Domain Foreign Agent. Σε αυτή την περίπτωση, ο προηγούμενος Domain Foreign Agent θα παίξει το ρόλο του “local home agent”. Στη συνέχεια, ο νέος Domain Foreign Agent θα στείλει το Registration reply στον mobile node και θα τον ενημερώσει για τον νέο του home agent.

Όπως βλέπουμε, η προσέγγιση αυτή, μειώνει ακόμη περισσότερο την κίνηση ελέγχου μεταξύ mobile node και home agent, όπως επίσης μειώνει κατά πολύ και την πιθανότητα απώλειας πακέτου. Το μειονέκτημα όμως είναι ότι απαιτεί τη δυνατότητα εντοπισμού θέσης από τους agents.

3.4.5 Multicast-Based Handover

Η πρόταση αυτή για fast handover [19], υιοθετεί την ιεραρχική αρχιτεκτονική των foreign agents, με τον Domain Foreign Agent να βρίσκεται στην κορυφή του Foreign Agent δένδρου, όπως περιγράφηκε σε προηγούμενη ενότητα. Η βασική ιδέα, βρίσκεται στην εκπομπή multicast πακέτων για να επιτευχθεί το fast handover.

Στο σχήμα 3-12, φαίνεται η ιεραρχική δομή των foreign agents. Οι foreign agents, υποτίθεται ότι έχουν buffers, και έχουν τη δυνατότητα να εγγραφούν σε multicast ομάδες. Η βασική λειτουργία, έχει ως εξής. Όταν ο mobile node καταχωρείται σε έναν Domain Foreign Agent, ο τελευταίος του αποδίδει μία μοναδική multicast IP διεύθυνση, και απαιτεί από τον foreign agent του mobile node, να εγγραφεί σε μια multicast ομάδα. Αυτός ο foreign agent, ενημερώνει τους γειτονικούς agent για να εγγραφούν στην ίδια multicast ομάδα. Κάθε φορά που ο Domain Foreign Agent αποστέλλει ένα πακέτο στον agent του mobile node, μέσω τεχνικών multicast στέλνει το πακέτο και στους υπόλοιπους agents, οι οποίοι είναι καταχωρημένοι στην multicast ομάδα.



Σχήμα 24

Μόνο ο agent που εξυπηρετεί τον mobile node θα στείλει το πακέτο στον node. Οι υπόλοιποι απλώς θα κρατήσουν τα πακέτα στον buffer, ώστε σε περίπτωση που

έχουμε handover, να ελαττωθεί η απώλεια πακέτων. Ας δούμε όμως τώρα, τι ακριβώς γίνεται όταν έχουμε το handover.

Μόλις ο mobile node, μετακινηθεί σε γειτονικό agent, στέλνει ένα “greet” (χαιρετιστήριο) μήνυμα στο νέο agent. Μέσα στο μήνυμα αυτό, εμπεριέχονται η multicast IP διεύθυνση του mobile node, η IP διεύθυνση του προηγούμενου agent, όπως επίσης και το ID του τελευταίου πακέτου που έλαβε ο mobile node. Το τελευταίο γίνεται, για να αποφευχθεί η αποστολή διπλών πακέτων προς τον mobile node. Εν συνεχεία, ο νέος agent ενημερώνει τον προηγούμενο ότι έχει γίνει handover, έτσι ώστε με τη σειρά του αυτός να ενημερώσει τους γειτονικούς τους agents να διαγραφούν από την εν λόγω multicast ομάδα. Από την άλλη πλευρά, ο νέος foreign agent, ενημερώνει τους γειτονικούς του agent να καταχωρηθούν στην multicast ομάδα.

Ο multicast-based handover μηχανισμός που περιγράψαμε παραπάνω, ελαττώνει κατά πολύ την πιθανότητα να έχουμε απώλεια πακέτων. Το πλεονέκτημα του σε σχέση με τον αλγόριθμο Position Leverage είναι ότι ο Domain Foreign Agent δεν είναι επιφορτισμένος με το δύσκολο έργο της διατήρησης και διαχείρισης των διευθύνσεων αλλά και των τοποθεσιών των mobile nodes. Αυτό που κάνει μόνο είναι να κατανείμει τις multicast διευθύνσεις. Μετά το αρχικό registration του mobile node στον Domain Foreign Agent, δεν υπάρχει καθόλου κίνηση σηματοδοσίας ανάμεσα σε αυτούς τους δύο, όσο ο mobile node κινείται στα όρια του συγκεκριμένου domain. Το μειονέκτημα που παρουσιάζει η συγκεκριμένη πρόταση, είναι ότι απαιτεί επιπρόσθετες λειτουργίες από τους agent αλλά επίσης παρουσιάζεται αυξημένη κίνηση μέσα στο domain λόγω του multicasting.

4 Βελτιστοποίηση της απόδοσης του Handover με τη χρήση Link Layer triggers.

Μέχρι εδώ, έχουμε παρουσιάσει τους τρόπους με τους οποίους ανιχνεύεται αλλά και εκτελείται ένα handover. Είδαμε ότι οι πληροφορίες στις οποίες βασίζονται οι διάφοροι αλγόριθμοι για την ανίχνευση του handover, αντλούνται αποκλειστικά και μόνο από το Network layer (ή L3), δηλαδή από τα advertisement μηνύματα, με εξαίρεση τον Parametric Cell Switching αλγόριθμο. Έχει γίνει όμως κατανοητό, ότι για τη βελτιστοποίηση της απόδοσης του handover initiation, απαιτείται η χρήση πληροφοριών που εξάγονται από το επίπεδο δικτύου. Στο κεφάλαιο αυτό, θα αναπτύξουμε τη «φιλοσοφία» γύρω από τη χρήση link layer (ή L2 layer) triggers όσον αφορά το handover.

Η χρήση L2 triggers μειώνει κατά πολύ το χρόνο που απαιτείται για την ανίχνευση αλλά και την εκκίνηση του handover. Πληροφορία, όπως η ισχύς του σήματος (που είναι μέγεθος από το link layer) μπορεί να είναι διαθέσιμη συνέχεια, συνεπώς να δίνει τη δυνατότητα στον mobile node να έχει μία εκτίμηση της ζεύξης ανά πάσα στιγμή. Προχωρώντας λίγο παραπέρα, ένας αλγόριθμος βασιζόμενος σε L2 πληροφορίες, μπορεί να ανιχνεύσει την απώλεια της σύνδεσης γρηγορότερα σε σχέση με έναν αλγόριθμο που βασίζεται μόνο στα advertisement μηνύματα. Επίσης, λαμβάνοντας υπόψιν L2 πληροφορίες, μπορούμε να ανιχνεύσουμε την εξασθένηση της ζεύξης και να προβούμε σε handover πριν η ζεύξη καταρρεύσει.

Η χρήση όμως L2 triggers παραβιάζει μία γενική αρχή των πρωτοκόλλων του Internet. *Το IP σχεδιάστηκε για να είναι ανεξάρτητο από τις υποκείμενες τεχνολογίες.* Η επιτυχία του Διαδικτύου όμως οφείλεται κατά ένα μεγάλο μέρος σε αυτή την αρχή. Αυτή η αρχή όμως έχει σαν αποτέλεσμα, στα IP δίκτυα να έχουμε μεταφορά IP πακέτων χωρίς να έχουμε τη μέγιστη απόδοση ή το ελάχιστο κόστος. Φαίνεται καθαρά δηλαδή, ότι η έρευνα και η ανάπτυξη στα IP πρωτόκολλα μη λαμβάνοντας υπόψιν τη τηλεπικοινωνιακή υποδομή των δικτύων πάνω στα οποία θα εφαρμοσθούν τα IP πρωτόκολλα, δε θα φέρει τα επιθυμητά αποτελέσματα. Για να δώσουμε ένα

παράδειγμα, στα IP δίκτυα όπου το μέσο μετάδοσης είναι ασύρματο, η χρήση των παραδοσιακών IP πρωτοκόλλων καταλήγει σε πολύ φτωχές επιδόσεις.

Το πάντρεμα του IP και των υποκείμενων τεχνολογιών αλλά και η χρήση πληροφοριών από το επίπεδο δικτύου στο IP επίπεδο είναι ένα ανοιχτό ερώτημα το οποίο συζητείται στην κοινότητα της IETF. Καθώς είναι κοινά αποδεκτό ότι η χρήση L2 πληροφοριών θα οδηγήσει σε αποδοτικότερη μεταφορά IP πακέτων, παρόλα αυτά υπάρχει ο φόβος ότι αυτό θα οδηγήσει στην εξάρτηση του IP επιπέδου από τις υποκείμενες τεχνολογίες. Εάν λάβουμε υπόψιν την ποικιλία των διάφορων τεχνολογιών και την εξάρτηση του IP σε κάθε τεχνολογία, αυτό θα οδηγήσει σε ένα μεγάλο αριθμό από συγκεκριμένες στοίβες πρωτοκόλλων. Όπως εύκολα μπορούμε να καταλάβουμε, αυτό θα εμποδίσει την ανάπτυξη και εφαρμογή των ετερογενών all-IP ασύρματων δικτύων.

Άρα το πρόβλημα που πρέπει να λυθεί είναι να ανεξαρτητοποιηθεί το IP όσο γίνεται από την υποκείμενη τεχνολογία. Για να γίνει αυτό, πρέπει το link-layer trigger να θεωρηθεί ως μία ειδοποίηση ότι ένα γεγονός έχει συμβεί ή πρόκειται να συμβεί. Ακολουθώντας αυτό τον ορισμό, μπορούμε να πούμε ότι ο trigger αποτελείται από τρεις συνιστώσες: το γεγονός που προκάλεσε τον trigger, την οντότητα που δέχεται τον trigger και την παράμετρο που παραδίδεται με τον trigger. Και οι τρεις αυτές συνιστώσες καλύπτονται σε αυτό το κεφάλαιο.

4.1 Link Layer Triggers

Στο σημείο αυτό, θα αναφέρουμε ποια μεγέθη μπορούν να υλοποιήσουν τους link-layer triggers.

Ισχύς Σήματος

Η ισχύς του σήματος είναι ένα μέγεθος που εκφράζει το πόσο ισχυρό είναι το σήμα που δέχεται ο δέκτης. Συχνά, αυτή η παράμετρος αναφέρεται και ως *Received*

Signal Strength Indicator (RSSI). Η ισχύς του σήματος στο δέκτη εκφράζει την εξασθένηση που υφίσταται το σήμα μέχρι να φτάσει στο δέκτη. Η εξασθένηση αυτή μπορεί να οφείλεται σε φυσικά φαινόμενα όπως π.χ. στην ανάκλαση, στην σκέδαση και στη διάθλαση.

Για να χρησιμοποιήσουμε την ισχύ του σήματος για handover trigger, πρέπει να λάβουμε υπόψιν τα παρακάτω:

- Η χρήση της ισχύς του σήματος για handover trigger απαιτεί τον καθορισμό ενός κατωφλίου. Εάν η ισχύς είναι μεγαλύτερη του κατωφλίου τότε η ζεύξη θεωρείται καλή. Διαφορετικά, η ζεύξη θεωρείται κακή και θα γίνει handover. Όμως, το κατώφλι της ισχύς του σήματος είναι μία παράμετρος που περιγράφεται στην εκάστοτε τεχνολογία. Για παράδειγμα, για το IEEE 802.11a με ρυθμό δεδομένων 54 Mbps το κατώφλι είναι -65 dBm ενώ για το IEEE 802.11b με ρυθμό δεδομένων 2 Mbps το κατώφλι είναι -70 dBm. Επί πλέον το κατώφλι μπορεί να είναι διαφορετικό για συσκευές διαφορετικών κατασκευαστών.
- Η μέτρηση της ισχύς του σήματος συνήθως περιλαμβάνει και την παρεμβολή, είτε γειτονικού καναλιού είτε ενδοκαναλική. Συνεπώς, σε περιβάλλον με υψηλή παρεμβολή παρόλο που το κανάλι δεν είναι κατάλληλο η μέτρηση θα μας δείχνει ένα ικανοποιητικό κανάλι.
- Η μέτρηση της ισχύς του σήματος απαιτεί από τον κινητό σταθμό να λαμβάνει τουλάχιστον ένα σήμα συνεχώς από το σταθμό βάσης. Στο W-CDMA για παράδειγμα, ο σταθμός βάσης εκπέμπει ένα πιλοτικό σήμα συνέχεια. Στο IEEE 802.11 όμως, ο σταθμός βάσης εκπέμπει ένα beacon σήμα το οποίο στέλνεται ανά 100 msec.
- Μία άλλη παράμετρος που πρέπει να λάβουμε υπόψιν, είναι ότι η λαμβανόμενη ισχύς του σήματος πολλές φορές μπορεί να ελαττωθεί απότομα. Ένα πολύ γνωστό παράδειγμα είναι το *corner-effect*. Όταν ο κινητός σταθμός κινείται σε ένα περιβάλλον Manhattan-like και όπως κινείται, στρίψει σε μία

γωνία, τότε η ισχύς τους λαμβανόμενου σήματος πέφτει απότομα.

- Επίσης, σημαντικό ρόλο παίζει και η κεραία που υπάρχει στον κινητό σταθμό. Με μία κεραία υψηλού κέρδους, η λαμβανόμενη ισχύς σήματος είναι μεγαλύτερη σε σχέση με τη λαμβανόμενη ισχύ που δίνει μία κεραία χαμηλότερου κέρδους. Εφαρμόζοντας το ίδιο κατώφλι και για τις δύο περιπτώσεις είναι πολύ πιθανό να έχουμε ανεπιθύμητα handover.

Λόγος Σήματος προς Παρεμβολή (SIR)

Ο λόγος σήματος προς παρεμβολή (Signal-to-Interference Ratio ή SIR) είναι ένα μετρήσιμο μέγεθος που μας δείχνει το πόσο υψηλότερη είναι η ισχύς του σήματος προς την ισχύ της παρεμβολής. Ως πηγές παρεμβολής είναι άλλοι κινητοί σταθμοί που εκπέμπουν στην ίδια συχνότητα, οπότε και μιλάμε για ενδοκαναλική παρεμβολή, ή σε γειτονική συχνότητα, όπου εδώ έχουμε την παρεμβολή γειτονικού καναλιού. Το SIR μετριέται σε decibel (dB).

Παρόμοια και με την ισχύ του σήματος, το SIR είναι ένα χρονικά μεταβαλλόμενο μέγεθος. Η λαμβανόμενη ισχύς αλλά και η παρεμβολή μεταβάλλεται με το χρόνο. Συνεπώς, πρέπει να υπολογίζεται η μέση τιμή του SIR. Συγκριτικά όμως με τη λαμβανόμενη ισχύ του σήματος, το SIR δίνει μια πιο ρεαλιστική εικόνα της ποιότητας της ζεύξης και αυτό διότι περιλαμβάνει την παρεμβολή.

Bit Error Rate (BER)

Το Bit error Rate ή αλλιώς ο ρυθμός λανθασμένων bit, είναι το ποσοστό των λανθασμένων bit ως προς το σύνολο των bit που έλαβε ο δέκτης. Το BER εκφράζεται συνήθως ως αρνητική δύναμη του δέκα.

Στα περισσότερα ασύρματα δίκτυα, χρησιμοποιούνται στο δέκτη, τεχνικές ελέγχου σφάλματος για την ανίχνευση αλλά και τη διόρθωση αυτών. Το BER μετριέται είτε πριν είτε μετά την διόρθωση σφαλμάτων. Όπως είναι λογικό, το BER είναι

διαφορετικό σε κάθε περίπτωση.

Frame Error Rate (FER)

Το Frame error Rate (FER) είναι το ποσοστό των λανθασμένων πλαισίων ως προς το συνολικό αριθμό των πλαισίων που έλαβε ο δέκτης. Όπως και το BER έτσι και το FER εκφράζεται ως αρνητική δύναμη του δέκα.

Σε μερικά πρωτόκολλα ασύρματης επικοινωνίας, οι κινητοί σταθμοί μπορούν να ανιχνεύσουν λανθασμένα ή χαμένα πλαίσια. Για παράδειγμα, στο IEEE 802.11, έχουμε την τεχνική του *send & wait* με άμεσο acknowledgement οπότε η ανίχνευση χαμένου πλαισίου είναι εύκολη υπόθεση. Γενικότερα, τα ARQ πρωτόκολλα όπου έχουμε επανάληψη εκπομπής, παρέχουν τον αριθμό των επαναλήψεων οπότε μπορούμε να υπολογίσουμε το FER.

Η αποτίμηση του FER απαιτεί την παρακολούθηση ενός σχετικά μεγάλου αριθμού πλαισίων. Όταν όμως έχουμε μία ζεύξη η οποία εξασθενεί πολύ γρήγορα, πολλά πακέτα τα οποία έχουν σταλθεί, δεν έχουν ληφθεί από τον δέκτη. Αυτό έχει ως αποτέλεσμα, να μην προσμετρώνται στο FER, οπότε το FER μας δείχνει ένα ικανοποιητικό κανάλι, ενώ δεν υπάρχει ουσιαστικά ζεύξη.

Συνοψίζοντας ως εδώ, έχουμε παρουσιάσει τις παραμέτρους που μπορούν να αποτελέσουν τους L2 triggers. Αυτοί είναι η ισχύς του σήματος, ο λόγος σήματος προς παρεμβολή, το bit error rate και το frame error rate. Η ισχύς του σήματος είναι μία ένδειξη για την ποιότητα της ζεύξης, απαιτεί όμως τον υπολογισμό του μέσου όρου για να αντισταθμίσει τις μικρής διάρκειας (short-term) διακυμάνσεις. Ωστόσο, η ισχύς του σήματος είναι το επιθυμητό σήμα εμπεριέχοντας τις παρεμβολές. Οπότε, το SIR είναι καταλληλότερο μέγεθος, αφού δίνει μια εικόνα της ισχύς του επιθυμητού σήματος και ξεχωριστά της παρεμβολής. Όμως, όπως αναφέραμε και παραπάνω, οι παραπάνω δύο παράμετροι απαιτούν τη συνεχή εκπομπή ενός σήμάτος από τους σταθμούς βάσης. Οι τεχνολογίες IS-95 και W-CDMA παρέχουν ένα τέτοιο σήμα, το λεγόμενο pilot signal. Σε

άλλες τεχνολογίες, όπως το IEEE 802.11 ή το Bluetooth χρησιμοποιούνται άλλοι μηχανισμοί, όπως η περιοδική εκπομπή beacon σημάτων.

Κάνοντας μία σύγκριση των παραμέτρων που παρουσιάσαμε, βλέπουμε ότι το SIR ενεργεί γρηγορότερα σε σχέση με το FER σε ένα κανάλι που έχουμε γρήγορες εναλλαγές. Επίσης, αν συνδυάσουμε την ισχύ του σήματος με το BER, μπορούμε να υπερκεράσουμε την αδυναμία της ισχύς σήματος, το ότι δηλαδή δεν λαμβάνει υπόψιν του την παρεμβολή, δηλαδή την ποιότητα της ζεύξης.

Εν τέλει, όλες αυτές οι παράμετροι είναι συγκεκριμένες για κάθε τεχνολογία ξεχωριστά. Μία λύση σε αυτό το πρόβλημα, είναι η κάθε παράμετρος να σταθμίζεται με έναν παράγοντα ο οποίος θα είναι ξεχωριστός για την εκάστοτε τεχνολογία. Αυτή η παραμετροποίηση των παραμέτρων κάνει δυνατή την βέλτιστη εκκίνηση του handover σε ετερογενής τεχνολογίες. Στο σχήμα 25 δίνουμε μία σύνοψη των διαθέσιμων παραμέτρων σε διάφορες ασύρματες τεχνολογίες.

L2 trigger	**IEEE 802.11b**	**Bluetooth**	**GSM**	**IS-95**	**W-CDMA**	**HIPERLAN/2**
Signal Strength Downlink	Yes (beacon)	Yes (beacon)	Yes (Control Channel)	Yes (Pilot Channel)	Yes (Pilot Channel)	Yes (beacon)
Signal Strength Uplink	No	NA	Yes (Control Channel)	No	Yes (Pilot Channel)	Yes (Via Resource Request)
SIR Downlink	Yes (Indirect)	Yes (Indirect)	No	No	Yes	Yes (Indirect)
SIR Uplink	No	NA	No	Yes	Yes	Yes
BER	No	No	Yes	No	No	No

Downlink			(Dedicated Mode)			
BER Uplink	No	NA	Yes (Dedicated Mode)	No	No	No
FER Downlink	Yes	Yes	NA	No	Yes	Yes
FER Uplink	No	NA	NA	No	Yes	Yes

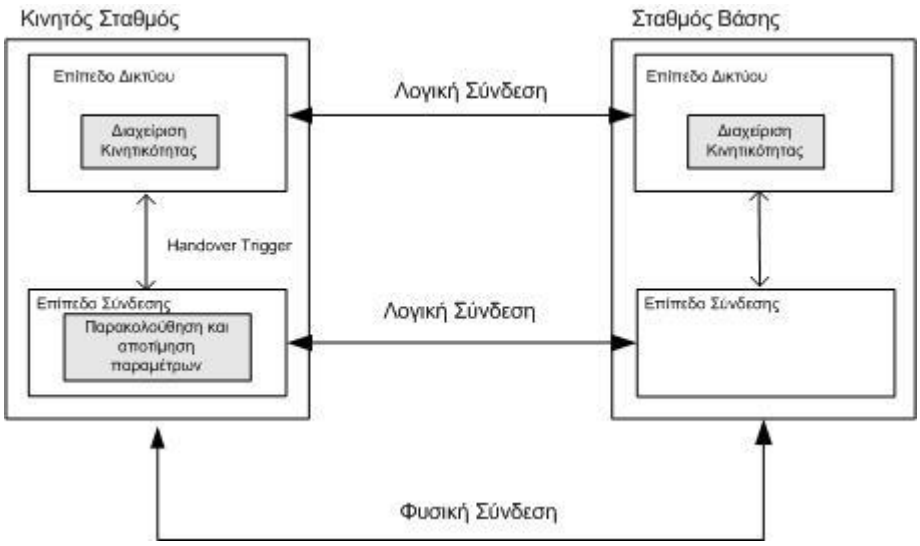
Σχήμα 25

4.2 Αλληλεπίδραση μεταξύ Επιπέδου Σύνδεσης και Επιπέδου Δικτύου

Στην προηγούμενη ενότητα, περιγράψαμε τις παραμέτρους που μπορούν να χρησιμοποιηθούν για L2 triggers. Στη συνέχεια θα περιγράψουμε την αλληλεπίδραση μεταξύ του επιπέδου σύνδεσης και του επιπέδου δικτύου.

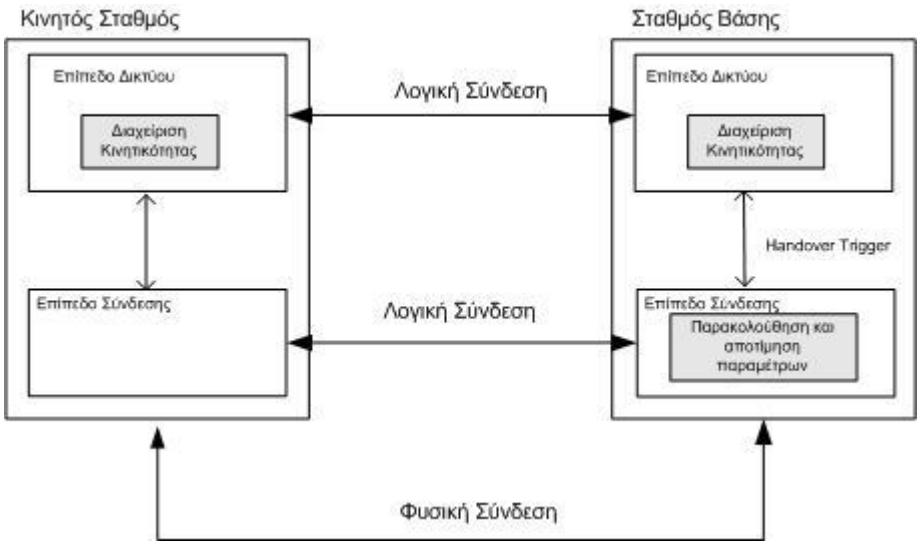
Κυρίως, τρεις περιπτώσεις handover με χρήση L2 triggers μπορούμε να διακρίνουμε:

Mobile triggered and initiated handover: Σε αυτή την περίπτωση, ο handover trigger δημιουργείται από το επίπεδο σύνδεσης στον κινητό σταθμό και εν συνεχεία μεταφέρεται στο επίπεδο δικτύου του κινητού σταθμού (vertical triggering).



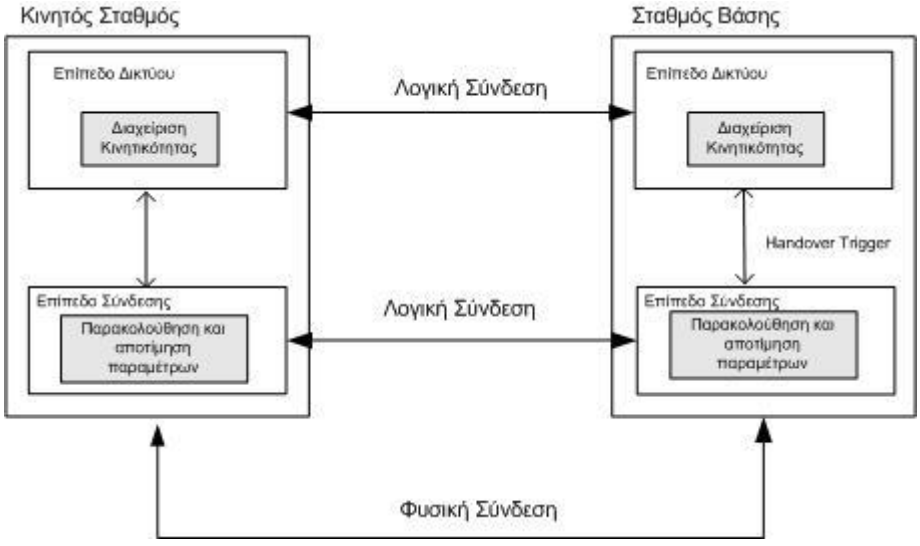
Σχήμα 26

Network triggered and initiated handover: Εδώ, ο handover trigger μεταφέρεται κάθετα (vertical triggering) παρόμοια με την προηγούμενη περίπτωση, όμως ο trigger δημιουργείται από το επίπεδο σύνδεσης του σταθμού βάσης και λαμβάνεται στο επίπεδο δικτύου του ίδιου.



Σχήμα 27

Mobile triggered and network initiated handover: Στην περίπτωση αυτή, ο handover trigger παράγεται από το επίπεδο σύνδεσης στον κινητό σταθμό και εν συνεχεία μεταφέρεται στο επίπεδο δικτύου του σταθμού βάσης, όπου αυτός αναλαμβάνει την εκκίνηση του handover.



Σχήμα 28

Για να γίνουν κατανοητά τα παραπάνω, υποθέτουμε ένα IEEE 802.11 ασύρματο δίκτυο, όπου ως L2 trigger, ορίζουμε την ισχύ του σήματος.

Στην περίπτωση *mobile triggered and network initiated handover,* το πρόγραμμα οδήγησης της ασύρματης κάρτας δικτύου στον κινητό σταθμό, παρακολουθεί την ισχύ του λαμβανόμενου σήματος. Αυτό επιτυγχάνεται με την παρακολούθηση των beacons που στέλνει ο σταθμός βάσης, περίπου ανά 100 msec. Να σημειώσουμε εδώ ότι οι σταθμοί βάσης διαφοροποιούνται από την MAC διεύθυνση που έχει ο καθένας.

Όταν η ισχύ του σήματος πέσει κάτω από ένα προκαθορισμένο κατώφλι, το επίπεδο σύνδεσης ενημερώνει το επίπεδο δικτύου ότι το κανάλι επικοινωνίας έχει χαθεί. Υπενθυμίζουμε ότι όλα αυτά γίνονται στον κινητό σταθμό. Στη συνέχεια, μόλις ο κινητός σταθμός λάβει ένα νέο beacon σήμα, τότε το επίπεδο σύνδεσης ενημερώνει το επίπεδο δικτύου για τη διαθεσιμότητα του νέου σταθμού βάσης, οπότε εν συνεχεία ο κινητός σταθμός εκκινεί τη διαδικασία του handover.

Στη δεύτερη περίπτωση *network triggered and initiated handover,* ο σταθμός βάσης παρακολουθεί την ισχύ του σήματος που λαμβάνει από τον κινητό σταθμό. Όταν ο τελευταίος πρόκειται να βγει εκτός των ορίων κάλυψης του σταθμού βάσης, η λαμβανόμενη ισχύς στον σταθμό βάσης πέφτει κάτω από το κατώφλι. Οπότε, όμοια με τη διαδικασία που περιγράψαμε παραπάνω, ενημερώνεται το επίπεδο δικτύου του σταθμού βάσης για το γεγονός και έτσι ο σταθμός βάσης αναγκάζει τον κινητό σταθμό να προβεί σε handover.

Τέλος, στην τρίτη περίπτωση *mobile triggered and network initiated handover* η διαδικασία έχει ως εξής. Ο κινητός σταθμός παρακολουθεί τη λαμβανόμενη ισχύ από το σταθμό βάσης. Εάν η ισχύς του σήματος πέσει κάτω από το κατώφλι, τότε ο κινητός σταθμός στέλνει ένα link layer μήνυμα στο επίπεδο σύνδεσης του σταθμού βάσης, όπου στο μήνυμα αυτό περιλαμβάνεται και η τιμή της λαμβανόμενης ισχύς στον κινητό σταθμό. Εν συνεχεία το επίπεδο σύνδεσης στο σταθμό βάσης μεταβιβάζει την πληροφορία αυτή στο επίπεδο δικτύου όπου στη συνέχεια εκκινούνται οι διαδικασίες για handover.

4.3 Υλοποίηση Low Latency Handover στο Mobile IP με τη χρήση L2 triggers.

Μέχρι εδώ έχουμε παρουσιάσει τη γενικότερη φιλοσοφία του cross layering, τους προβληματισμούς που απασχολούν την επιστημονική κοινότητα όσον αφορά την εφαρμογή αυτού, αλλά και το πως εννοείται το cross layering στο handover.

Στην ενότητα αυτή θα δούμε το πως μπορεί να χρησιμοποιηθούν οι L2 triggers στη διαδικασία του handover στο Mobile IP. Θα παρουσιάσουμε και θα αναλύσουμε μία τεχνική η οποία έχει προταθεί από τον K. El. Malki [14]. Η μέθοδος αυτή, δεν προτείνει κάποιον συγκεκριμένο L2 trigger, απλώς ως L2 trigger ορίζει κάθε πιθανό μέγεθος που παρουσιάσαμε παραπάνω.

Ο K. El. Malki περιγράφει τρεις τεχνικές για το handover. Την *Pre-Registration,* την *Post-Registration* και την *Combined* τεχνική.

Η Pre-Registration τεχνική επιτρέπει στον mobile node να εμπλακεί σε ένα προβλεπόμενο L3 handover πριν ολοκληρωθεί το L2 handover. Το L3 handover μπορεί να είναι είτε mobile-initiated είτε network initiated. Ανάλογα, οι L2 triggers μπορούν να υπάρχουν και στον mobile node αλλά και στον foreign agent. Δεν προτείνονται νέα μηνύματα, παρά μόνο μία επέκταση στο Agent Solicitation μήνυμα στην περίπτωση που έχουμε mobile-initiated handover.

Η Post-Registration τεχνική προτείνει κάποιες επεκτάσεις στο πρωτόκολλο του Mobile IP, έτσι ώστε οι oFA και nFA, κάνοντας χρήση L2 triggers να εγκαθιδρύσουν ένα BET μεταξύ αυτών των δύο. Το tunnel αυτό δημιουργείται για να μπορεί ο mobile node να χρησιμοποιεί τον oFA ενώ βρίσκεται στο δίκτυο του nFA. Η τεχνική αυτή εγγυάται συνεχή σύνδεση του mobile node με το δίκτυο, πράγμα το οποίο έχει θετικό αντίκτυπο στις real-time εφαρμογές.

Η Combined μέθοδος είναι η παράλληλη λειτουργία των δύο τεχνικών που αναφέραμε παραπάνω. Αν το Pre-Registration handover μπορεί να γίνει πριν ολοκληρωθεί το L2 handover, τότε θα γίνει Pre-Registration handover. Ωστόσο, εάν δεν μπορεί να γίνει Pre-Registration handover, τότε ο oFA αποστέλλει την κίνηση που προορίζεται για τον mobile node, προς τον nFA, έχουμε δηλαδή ένα Post-Registration

handover. Και οι τρεις αυτές τεχνικές περιγράφονται παρακάτω αναλυτικά. Πριν προχωρήσουμε όμως στην ανάλυση της κάθε τεχνικής, πρέπει να αναφέρουμε την ορολογία που θα χρησιμοποιήσουμε για την περιγραφή τους.

4.3.1 Ορολογία

oFA – Έτσι ορίζουμε τον προηγούμενο Foreign Agent, ο οποίος εμπλέκεται στην διαχείριση της Care-of-Address του mobile node, προτού γίνει το L3 handover.

nFA – Ο νέος Foreign Agent, ο οποίος προβλέπεται να αποδώσει την νέα Care-of-Address στον mobile node, αφού επιτευχθεί το L2 handover.

aFA – Ο Foreign Agent ο οποίος βρίσκεται στο τέρμα του BET στην Post-Registration τεχνική.

L2 handover – Η μετακίνηση της link layer σύνδεσης του mobile node από ένα ασύρματο access point σε ένα άλλο.

L3 handover – Η μετακίνηση του mobile node από έναν Foreign Agent σε έναν άλλο, όπου έχουμε και αλλαγή της Care-of-Address.

L2 trigger – Πληροφορία από το link layer προς το network layer όπου ενημερώνει το τελευταίο ότι κάποιο συγκεκριμένο γεγονός έχει συμβεί, πριν αλλά και μετά το L2 handover. Υπενθυμίζουμε εδώ ότι δεν αναφερόμαστε σε συγκεκριμένα μεγέθη.

L2-MT ή mobile trigger – Ένας L2 trigger ο οποίος λαμβάνει χώρα στον mobile node και ενημερώνει ότι υπάρχει κίνηση προς κάποιον nFA.

L2-ST ή source trigger – Ένας L2 trigger ο οποίος λαμβάνει χώρα στον oFA, και τον ενημερώνει ότι ένα L2 handover πρόκειται να συμβεί.

L2-TT ή target trigger – Ένας L2 trigger ο οποίος συμβαίνει στον nFA, και τον ενημερώνει ότι ένας mobile node πρόκειται να μεταβιβασθεί στον nFA.

L2-LU ή link up – Ένας L2 trigger ο οποίος λαμβάνει χώρα είτε στον mobile node είτε στον nFA και ενημερώνει ότι έχει επιτευχθεί L2 επικοινωνία μεταξύ mobile node

και foreign agent.

L2-LD ή link down – Ένας L2 trigger που συμβαίνει στον oFA και τον ενημερώνει ότι η L2 επικοινωνία μεταξύ mobile node και oFA έχει πέσει.

BET ή bi-directional edge tunnel – Είναι ένα δικατευθυντήριο tunnel που εγκαθιδρύεται μεταξύ δύο foreign agent με σκοπό την προσωρινή δρομολόγηση της κίνησης που αφορά τον mobile node, χωρίς να απαιτείται από τον τελευταίο να αλλάξει την Care-of-Address του.

Network-initiated handover – Είναι το L3 handover όπου η εκκίνηση του γίνεται είτε από τον oFA είτε από τον nFA.

Mobile-initiated handover – Είναι το L3 handover, όπου η εκκίνηση αυτού γίνεται από τον mobile node.

4.3.2 Pre-Registration Handover

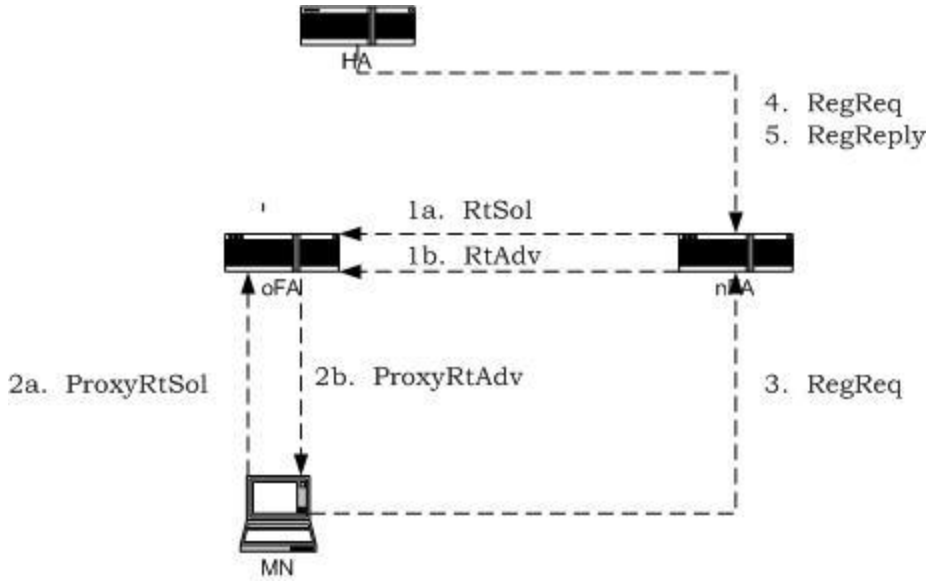
Η μέθοδος αυτή βασίζεται στην αρχική ιδέα του Mobile IP για το handover, όπως περιγράφεται στις προδιαγραφές του. Δηλαδή:

- Ο mobile node λαμβάνει advertisement μήνυμα από έναν νέο foreign agent.
- Το μήνυμα αυτό δηλώνει ότι ο mobile node έχει μεταβεί σε ένα νέο υποδίκτυο.
- Στη συνέχεια ο mobile node, μέσω των διαδικασιών που έχουμε περιγράψει, καταχωρείται στον νέο foreign agent.

Η Pre-Registration μέθοδος, κάνει χρήση L2 triggers είτε στον mobile node είτε στον foreign agent. Αυτό εξαρτάται από το αν λαμβάνει χώρα mobile ή network-initiated handover.

Λειτουργία

Ο μηχανισμός του Pre-Registration handover συνοψίζεται στο σχήμα 29.



Σχήμα 29

Παρακάτω, περιγράφουμε αναλυτικά τη λειτουργία του πρωτοκόλλου.

1. Το μήνυμα 1a είναι τύπου Router Solicitation (RtSol) από τον oFA προς τον nFA. Αυτά τα μηνύματα σκοπό έχουν να μειώσουν την καθυστέρηση του handover. Για αυτό, ο oFA πρέπει να αιτείται και να κρατάει σε μία cache, advertisement μηνύματα από τους γειτονικούς nFA. Όταν το L3 handover εκκινείται από έναν target L2 trigger στον nFA (L2-TT), το μήνυμα 1b ισούται με το μήνυμα 2b και στέλνεται αυτόκλητα στον MN (tunneled πάντα, μέσω του oFA).

2. Το 2a είναι ένα Proxy Router Solicitation (PrRtSol) μήνυμα. Είναι διαφορετικό από το κανονικό Agent Solicitation μήνυμα, μιας και στην πράξη ο MN αιτείται ένα advertisement μήνυμα από διαφορετικό agent από αυτόν που λαμβάνει το εν λόγω μήνυμα. Η παρουσία αυτού του μηνύματος υποδεικνύει ότι το handover είναι mobile-initiated, ενώ η απουσία αυτού του μηνύματος σημαίνει ότι έχουμε network-initiated handover. Στην περίπτωση του mobile-initiated handover, το μήνυμα 2a λαμβάνει χώρα εάν υπάρχει L2 trigger στον MN για να ζητήσει Proxy Router Advertisement μήνυμα. Όταν ο oFA λάβει ένα τέτοιο μήνυμα, τότε πρέπει οπωσδήποτε να απαντήσει στον MN

με Proxy Router Advertisement (2b) μήνυμα προς τον MN. Στο network-initiated source-triggered handover, L2 trigger έχουμε στον oFA ο οποίος πρέπει να στείλει Agent Advertisement μήνυμα στον MN χωρίς να περιμένει ο τελευταίος να το αιτηθεί.

3. Ο MN ανιχνεύει την κίνηση σύμφωνα με τα Agent Advertisement μηνύματα που λαμβάνει και αν χρειάζεται, στέλνει Registration Request (RegReq) (3) μήνυμα προς τον nFA. Το (3) δρομολογείται διαμέσου του oFA αφού ο MN δεν είναι απευθείας συνδεδεμένος με τον nFA, μέχρι να γίνει το L2 handover.

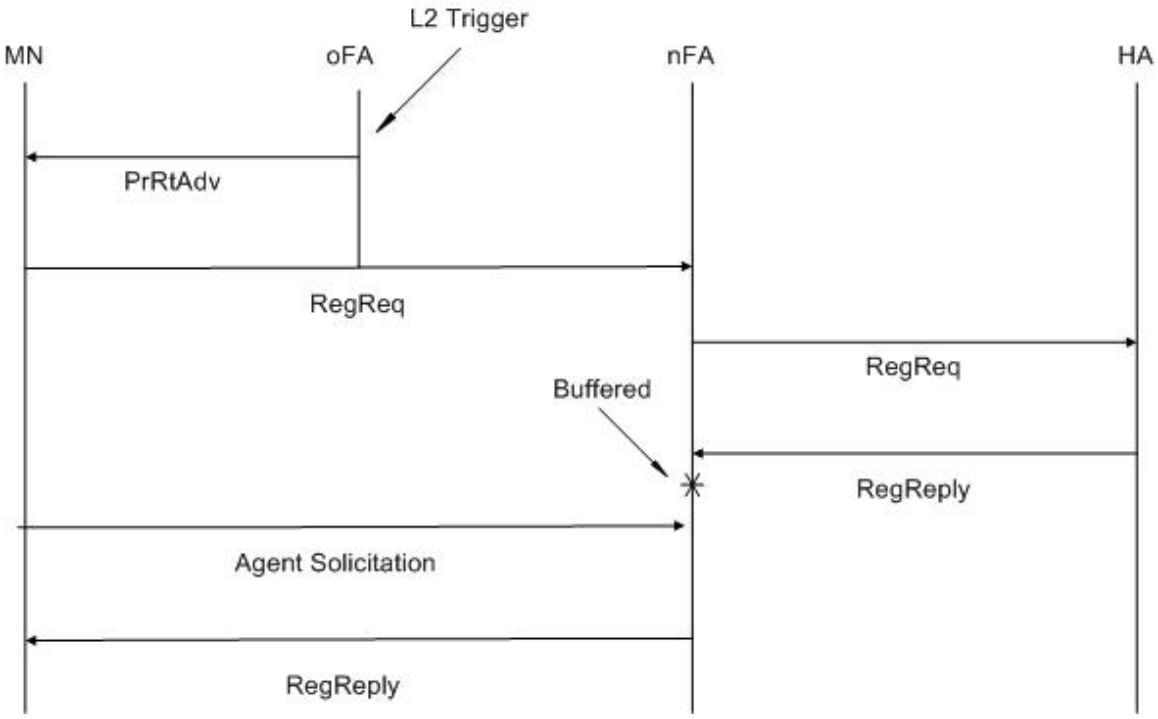
4. Τα μηνύματα 4 και 5 ολοκληρώνουν το γνωστό Mobile IP Registration.
Στην περίπτωση όπου ο MN δεν έχει συνδεθεί ακόμη στο nFA, το Registration Reply στο μήνυμα 5, πρέπει να γίνει buffered από τον nFA, έως ότου συνδεθεί ο MN και του αποσταλθεί το μήνυμα.

5. Αν το Registration είναι επιτυχημένο τότε τα πακέτα από τον HA ,μέσω tunneling τεχνικών, φθάνουν στον nFA και στη συνέχεια στον MN.

Συνοψίζοντας, με την PRE REGISTRATION τεχνική, πριν ακόμη χαθεί η L3 σύνδεση μεταξύ MN και oFA ο MN καταχωρείται στον nFA. Η χρονική στιγμή κατά την οποία ο L2 trigger εμφανίζεται στον oFA ή στον MN συγκρινόμενη με την χρονική στιγμή κατά την οποία πραγματοποιείται το L2 handover είναι πολύ σημαντικό σημείο, για να έχουμε τη βέλτιστη απόδοση του αλγορίθμου. Έτσι, στη βέλτιστη περίπτωση, ο L2 trigger θα ενεργοποιηθεί και έπειτα θα ολοκληρωθούν τα τέσσερα βήματα σηματοδοσίας που περιγράψαμε παραπάνω, πριν ακόμη ο MN κινηθεί. Δηλαδή το Registration Reply μήνυμα που έχει στείλει ο HA θα πρέπει να το λάβει ο MN τη στιγμή που L2 ζεύξη μεταξύ MN και nFA έχει ολοκληρωθεί. Έτσι δεν θα υπάρχει κάποιο πρόβλημα στο να ολοκληρωθεί το L3 handover.

Στη συνέχεια θα περιγράψουμε τα χρονικά διαγράμματα ανταλλαγής μηνυμάτων στις περιπτώσεις Network Initiated – Source Trigger, Network Initiated – Target Trigger και Mobile Initiated handover.

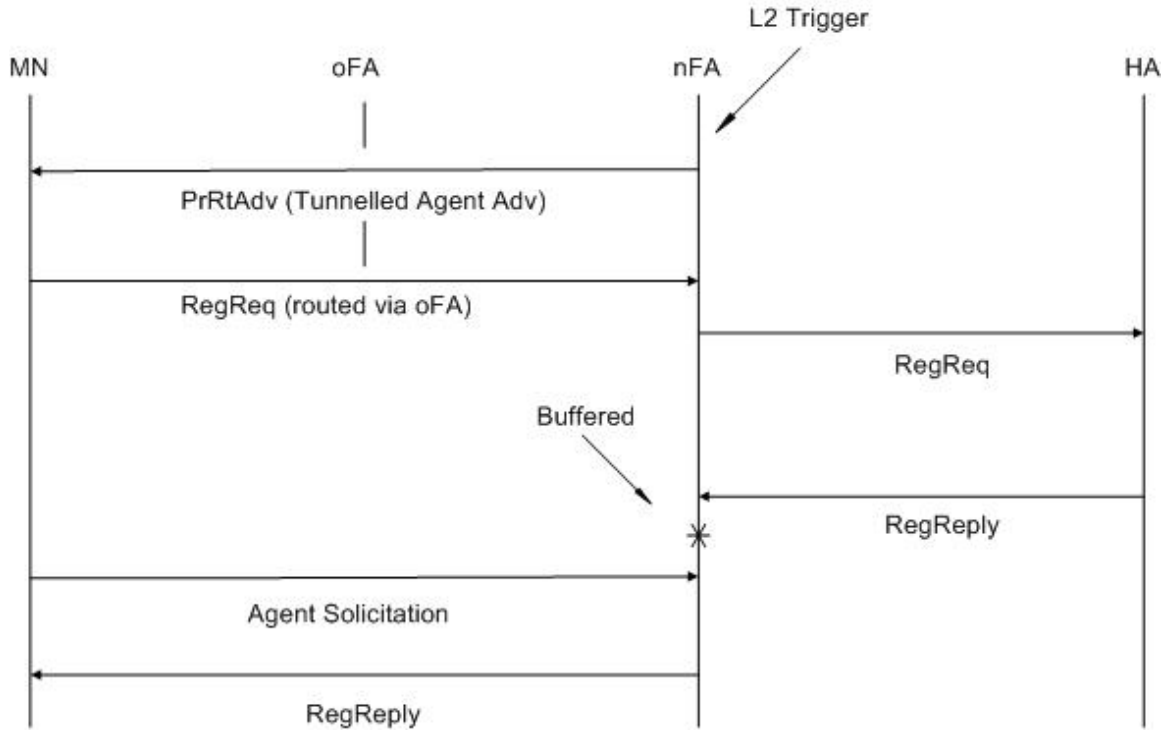
Network Initiated Handoff



Σχήμα 30

Εδώ έχουμε το χρονικό διάγραμμα μηνυμάτων στην περίπτωση Network Initiated, Source Trigger handoff. Πιο συγκεκριμένα, η διαδικασία του handover ξεκινά με την ύπαρξη ενός L2 trigger στον oFA οπότε αυτός αμέσως στέλνει στον MN ένα PrRtAdv μήνυμα. Δηλαδή τον ενημερώνει για την ύπαρξη του nFA. Οπότε με τη σειρά του ο MN στέλνει ένα RegReq μήνυμα στον nFA όπου ο τελευταίος με τη σειρά του το στέλνει στον HA. Το RegReply μήνυμα με το οποίο απαντά ο HA γίνεται buffered στον nFA. Όταν ο MN στείλει Agent Solicitation μήνυμα στον nFA αυτός με τη σειρά του, του απαντά με το RegReply το οποίο έχει καταχωρημένο.

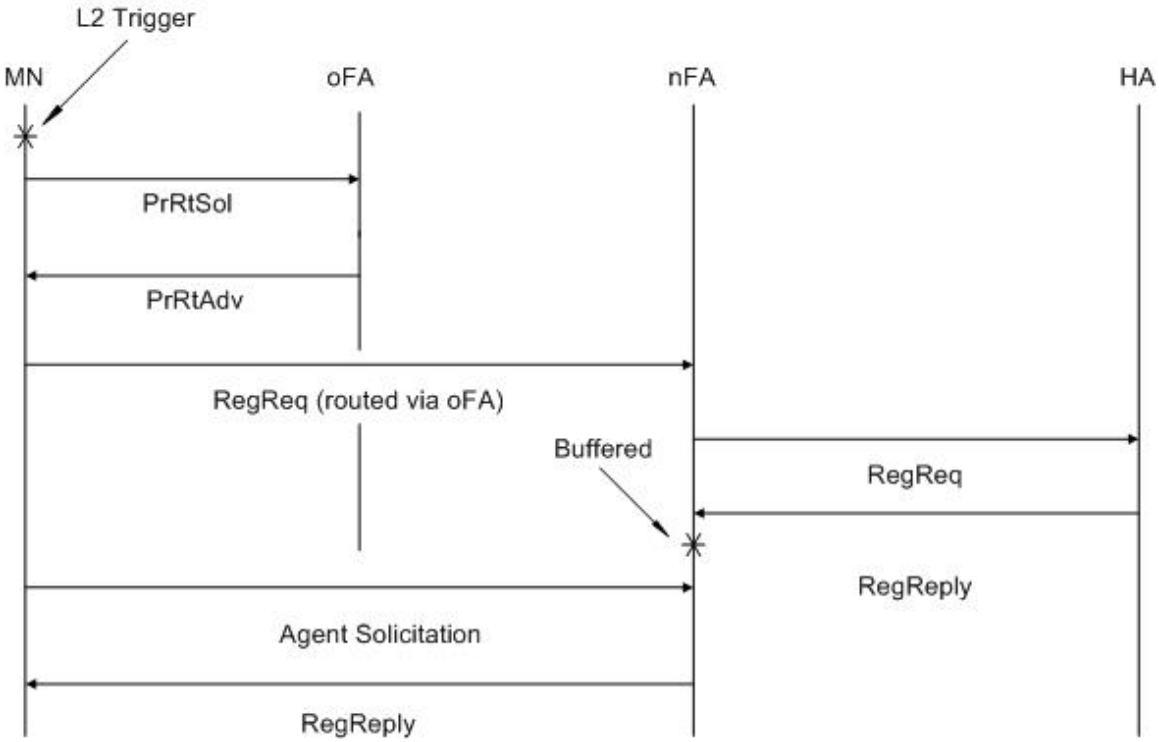
Άρα καταλαβαίνουμε ότι πολύ σημαντικό κομμάτι όλης αυτής της διαδικασίας είναι ο nFA να έχει καταχωρημένο το RegReply μήνυμα πριν αποπειραθεί ο MN να προβεί σε L3 handover.



Σχήμα 31

Εδώ έχουμε την περίπτωση Network Initiated, Target Trigger Handover. Δηλαδή την διαδικασία του handover την εκκινεί ο nFA. Εδώ έχουμε L2 trigger στον nFA. Δηλαδή, για παράδειγμα, στην περιοχή κάλυψής του αντιλαμβάνεται την ύπαρξη ενός MN. Τότε στέλνει ένα PrRtAdv μήνυμα μέσω του oFA στον MN. Στην συνέχεια απαντά ο MN με Registration Request μήνυμα στον nFA , πάντα με τεχνικές tunneling διαμέσου του oFA. Ο nFA με τη σειρά του στέλνει το μήνυμα αυτό στον HA και ακολουθείται η διαδικασία που περιγράψαμε και στην προηγούμενη περίπτωση.

Mobile Initiated Handover



Σχήμα 32

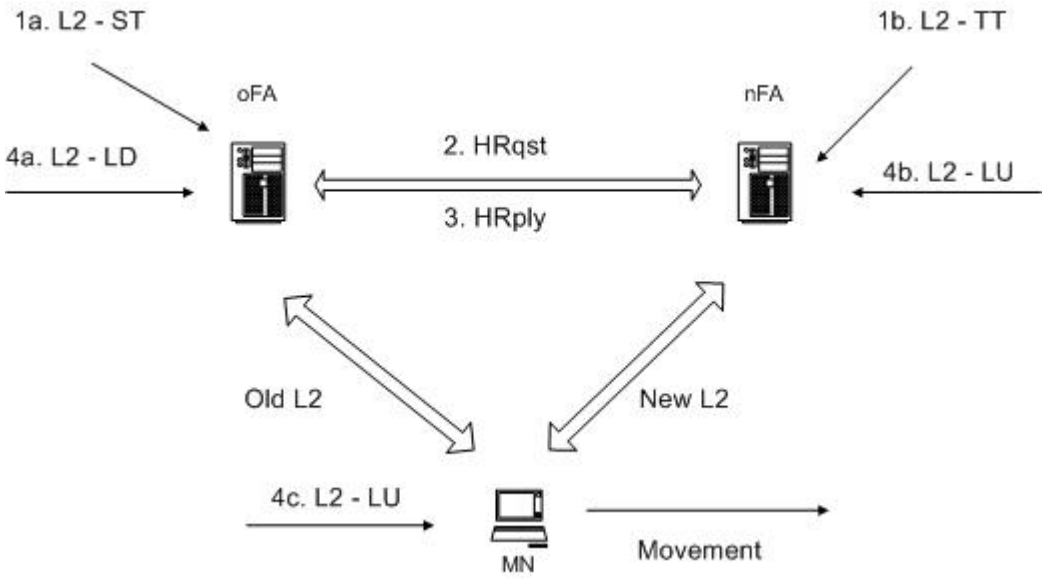
Στο παραπάνω σχήμα έχουμε την περίπτωση του Mobile Initiated Handover. Πιο αναλυτικά, το handover εκκινείται όταν ο MN δεχθεί ένα L2 trigger, πληροφορώντας τον ότι σύντομα θα μετακινηθεί σε έναν νέο nFA. Στη συνέχεια στέλνει ένα PrRtSol μήνυμα στον oFA. Ουσιαστικά αιτείται στον oFA, ένα PrRtAdv μήνυμα. Να σημειώσουμε εδώ ότι στον L2 trigger περιλαμβάνεται και ένα αναγνωριστικό για τον nFA. Για παράδειγμα, εάν έχουμε ένα Wireless LAN , ο MN γνωρίζει την IP διεύθυνση του nFA στον οποίο θα μεταβεί, λόγω του ότι η ισχύς του σήματος του συγκεκριμένου nFA αυξάνεται. Οπότε ο oFA στέλνει το κατάλληλο PrRtAdv μήνυμα το οποίο έχει καταχωρημένο και στη συνέχεια ακολουθείται η γνωστή διαδικασία.

Με τη βοήθεια των παραπάνω σχημάτων, περιγράψαμε τα τρία είδη handover που υλοποιούνται με την Pre – Registration τεχνική.

4.3.3 Post – Registration Handover

Η δεύτερη μέθοδος που προτείνει ο Malki, είναι η Post – Registration τεχνική. Η βασική ιδέα αυτής της τεχνικής, είναι ότι ο MN έχει μεταφερθεί σε περιοχή κάλυψης του nFA, έχει γίνει δηλαδή L2 handover, αλλά παρόλα αυτά είναι registered στον oFA. Δηλαδή η διαφορά με την προηγούμενη τεχνική, είναι στο ότι πρώτα ο MN έχει μετακινηθεί στον nFA και έπειτα εκκινούνται οι διαδικασίες για Mobile IP Registration. Να σημειώσουμε εδώ, ότι ο MN δεν εμπλέκεται καθόλου στην διαδικασία. Η μέθοδος Pre – Registration εισάγει δύο νέα μηνύματα. Το Handover Request (HRqst) και το Handover Reply (HRply). Αυτά τα δύο μηνύματα εκκινούν την διαδικασία για την εγκαθίδρυση ενός BET μεταξύ δύο Foreign Agents. Ας δούμε όμως παρακάτω την ανάλυση αυτής της τεχνικής.

Two Party Handover



Σχήμα 33

Στο παραπάνω σχήμα, περιγράφουμε την απλούστερη περίπτωση. Το two party handover. Δηλαδή όταν εμπλέκονται μόνο ο oFA και ο nFA. Πιο αναλυτικά:

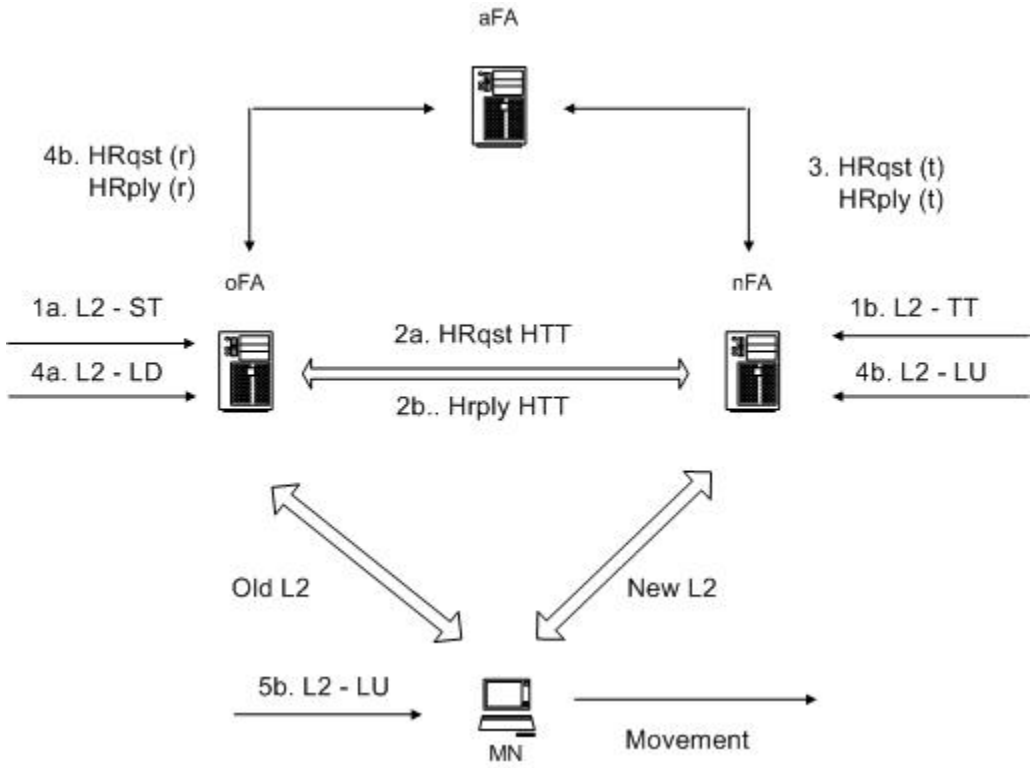
1. Ο oFA ή ο nFA δέχονται έναν L2 trigger πληροφορώντας ότι ένας συγκεκριμένος MN θα μετακινηθεί από τον oFA στον nFA. Υπάρχουν δύο υποπεριπτώσεις. Ο L2 trigger να είναι source trigger στον oFA ή να είναι target trigger στον nFA. Οπότε, ο FA ο οποίος δέχεται το L2 trigger στέλνει ένα HRqst μήνυμα στον άλλο FA για την ίδρυση ενός BET. Στην περίπτωση που ο oFA στέλνει το μήνυμα, τότε αυτό ονομάζεται HRqst(s). Στην αντίθετη περίπτωση, ονομάζεται HRqst(t).

Ο FA που λαμβάνει το μήνυμα, απαντά με HRply. Όπως περιγράψαμε και πριν, αν είναι απάντηση σε HRqst(s) τότε ονομάζεται HRply(s) ή διαφορετικά HRply(t) αν είναι απάντηση σε HRqst(t). Να σημειώσουμε εδώ, ότι στα HRqst και HRply μηνύματα που περιγράφουμε, εμπεριέχονται μεταξύ άλλων η διάρκεια ζωής του tunnel αλλά επίσης και στοιχεία για τον MN.

Στη συνέχεια, μόλις ο oFA δεχθεί ένα L2 – LD, πράγμα το οποίο σημαίνει ότι έχει χαθεί η ζεύξη με τον MN, ξεκινά την δρομολόγηση των πακέτων προς τον nFA, μέσω του BET. Με τη σειρά του ο nFA, μόλις λάβει L2 – LU, ξεκινά την δρομολόγηση των tunneled εισερχόμενων πακέτων προς τον MN αλλά και την δρομολόγηση των εξερχόμενων πακέτων με τεχνικές reverse tunneling. Μόλις ο MN λάβει L2 – LU trigger μπορεί να ξεκινήσει την γνωστή διαδικασία του Mobile Registration. Στην περίπτωση που ο MN μετακινηθεί προς τρίτο FA, πριν γίνει Registration, θεωρείται ότι είναι ακόμη registered στον oFA. Επίσης, στην περίπτωση του ping pong effect, αυτό μπορεί να γίνει αντιληπτό από τον oFA ή τον nFA από τους triggers που εμφανίζονται, (π.χ. όταν έχουμε L2 – LD και αμέσως μετά L2-LU). Οπότε σε αυτήν την περίπτωση, μπορεί να ακυρώσει το BET με ένα μήνυμα HRqst(r). Το μήνυμα αυτό ακυρώνει ένα ήδη υπάρχον BET.

Three Party Handover

Το Three party handover, είναι εφαρμόσιμο όταν ο MN ήδη λαμβάνει και στέλνει πακέτα μέσω του BET, αλλά δεν έχει γίνει registered στον nFA και κινείται προς έναν νέο Foreign Agent.



Σχήμα 34

Ο FA στον οποίο είναι registered ο MN, δηλαδή ο oFA, ονομάζεται aFA (anchor Foreign Agent). Στο παραπάνω σχήμα, περιγράφουμε την περίπτωση, όπου ο MN είναι συνδεδεμένος στο L2 επίπεδο με τον oFA, αλλά L3 σύνδεση έχει με τον aFA, δηλαδή έχει προηγηθεί Post Registration Handover. Άμεσα, πριν ακόμη ο MN εκκινήσει την διαδικασία του Mobile IP Registration, κινείται προς τον nFA. Οπότε, ο oFA ενημερώνει τον nFA ώστε ο τελευταίος να αιτηθεί για ένα BET με τον aFA. Αυτό γίνεται με τα HTT

μηνύματα. Ας δούμε όμως, πιο αναλυτικά τα βήματα

1) Ο oFA ή ο nFA, λαμβάνουν ένα L2 trigger όπου ειδοποιεί ότι ο συγκεκριμένος MN προκειται να μετακινηθεί. Υπάρχουν δύο περιπτώσεις.

   α) Να είναι L2 – ST στον oFA.

   β) Να είναι L2 –TT στον nFA.

2) Στη συνέχεια, oFA και nFA ανταλλάσσουν HTT/HRply ή HTT/HRqst μηνύματα

   α) Αν έχουμε L2 – ST trigger, τότε ο oFA στέλνει HTT μήνυμα στον nFA , το οποίο περιέχει την IP address του MN, την IP address του HA, την L2 address του MN και την IP address του aFA. Αυτή η πληροφορία είναι αρκετή έτσι ώστε ο nFA να εκτελέσει ένα target triggered handover. Ο nFA απαντά με ένα HRply (s) μήνυμα.

   β) Αν έχουμε L2 –TT trigger, τότε ο nF A στέλνει HRqst (t) μήνυμα στον oFA, σαν να είχαμε two party handover. Ο oFA απαντά με HTT μήνυμα εμπεριέχοντας τις ίδιες πληροφορίες όπως παραπάνω. Η πληροφορίες αυτές είναι αρκετές για τον nFA έτσι ώστε να προβεί σε target triggered handover με τον aFA.

3) Με τη λήψη του HTT, ο nFA τσεκάρει εάν ήδη στέλνει tunneled πακέτα στον MN. Αν ναι, τότε προχωράει στο βήμα 5. Αν όχι, τότε ο nFA εκτελεί ένα target triggered handover με τον aFA, ανταλλάσσοντας HRqst (t)/ HRply (t) μηνύματα.

4) Κατά τη διάρκεια του L2 handover, όπου ο MN δεν είναι συνδεδεμένος με κανέναν FA, ο aFA και ο oFA ανταλλάσσουν μηνύματα για να ακυρώσουν το BET μεταξύ τους, έτσι ώστε ο aFA να ιδρύσει ένα BET με τον nFA.

5) Την στιγμή που έχουμε ένα L2 – LU trigger στον nFA, αυτό σημαίνει ότι έχει γίνει το L2 handover. Οπότε:

   α) Ο nFA στέλνει τα πακέτα που προορίζονται για τον MN στον MN.

   β) Ο MN εκκινεί την γνωστή διαδικασία του Mobile IP Registration.

Ένα σημείο που αξίζει περαιτέρω προσοχή και μελέτη, είναι το πότε ακριβώς πρέπει να ξεκινήσει ο aFA να στέλνει encapsulated πακέτα προς τον nFA. Οι δύο ακραίες χρονικές στιγμές είναι α) μόλις έχει αποστείλει το HRply (t) μήνυμα στον nFA ή β) μόλις λάβει το HRqst (r) μήνυμα από τον oFA. Πρόσθετα, εάν επιλέξουμε την πρώτη περίπτωση, ο aFA μπορεί να στέλνει τα πακέτα και στον oFA μέχρι να λάβει το HRqst (r) μήνυμα από τον oFA. Σε αυτήν την περίπτωση όμως υπάρχει ο κίνδυνος ο MN να λάβει διπλά πακέτα. Στην δεύτερη περίπτωση προσθέτουμε μεγάλο latency. Υπάρχει πάντα όμως η δυνατότητα ο aFA να εκκινήσει το BET ανάμεσα στις δύο χρονικές στιγμές 1 και 2.

4.3.4 Combined Handoff Method

Η Combined Handoff Μέθοδος χρησιμοποιεί και την Pre-Registration τεχνική αλλά και την Post –Registration. Αν το Pre-Registration Handover δεν ολοκληρωθεί μέσα σε ένα συγκεκριμένο χρονικό διάστημα, τότε εκκινούνται οι διαδικασίες για Post-Registration Handover. Αυτή η μέθοδος προστατεύει τον MN από καθυστερήσεις που προέρχονται από την απώλεια του Mobile IP Registration Reply μήνυματος.

Όταν ο nFA λάβει ένα target trigger θα ακολουθήσει την Pre-Registration διαδικασία. Αυτόματα, εκκινείται και ένα χρονόμετρο (timer). Σύμφωνα με την Pre-Registration μέθοδο, ο nFA θα λάβει ένα Registration Request μήνυμα από τον MN. Στην περίπτωση που τη διαδικασία για το Handover δεν την έχει εκκινήσει ο nFA, (στην περίπτωση του Mobile initiated ή του Network initiated source-triggered handover, άρα δεν γνωρίζει ο nFA ότι έχει ξεκινήσει το handover), τότε αμέσως με τη λήψη του Registration Request μηνύματος ξεκινά ο timer.

Και στις δύο περιπτώσεις όμως, αμέσως με τη λήψη του Registration Reply μηνύματος από τον HA, ο timer μηδενίζεται. Στην περίπτωση όπου λήξει ο timer πριν ο nFA λάβει Registration Reply μήνυμα, τότε εκκινούνται οι διαδικασίες για Post-Registration Handover.

5. Απόδοση L2-L3 trigger handover αλγορίθμων

Η πλατφόρμα προσομοίωσης όπου έγινε η κατασκευή του ασύρματου δικτύου για την μέτρηση απόδοσης Layer2 και Layer 3 handover αλγόριθμων ονομάζεται Omnet ++ [20]. Το Omnet++ μοντελοποιεί ασύρματα και ενσύρματα δίκτυα και πρωτόκολλα με τη βοήθεια απλών και σύνθετων στοιχείων (modules). Τα απλά στοιχεία που υλοποιούν δικτυακές οντότητες ή μία συγκεκριμένη συμπεριφορά προσδιορίζονται από παραμέτρους και αναπαρίστανται σαν C++ κλάσεις. Τα σύνθετα στοιχεία απαρτίζονται απο πολλά συνήθως απλά στοιχεία είτε άλλα σύνθετα. Η επικοινωνία μεταξύ των στοιχείων γίνεται μέσω της ανταλλαγής μηνυμάτων. Η αποστολή και λήψη ενός τέτοιου μηνύματος διακρίνεται σαν συγκεκριμένο γεγονός.

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. Για την εν λόγω προσομοίωση, έγινε χρήση του Ipv6Suite Simulation Framework [21], μίας open-source επέκτασης, η οποία επιτρέπει την προσομοίωση Ipv6 πρωτόκολλων και δικτύων.

Να σημειώσουμε εδώ, ότι παρόλο που η εν λόγω προσομοίωση επικεντρώνεται στο MobileIPv6 πρωτόκολλο, οι μετρήσεις και τα αποτελέσματα αυτών ισχύουν και για το MobileIPv4 πρωτόκολλο.

5.1 Omnet++

Το OMNet++ είναι μία αντικειμενοστραφής πλατφόρμα προσομοίωσης η οποία αποτελείται απο ιεραρχικά δομημένα στοιχεία και παρέχει γραφικό περιβάλλον προσομοίωσης. Το OMNeT χρησιμοποιεί γεγονότα για την γνωστοποιηση ενός συμβάντος. Τα στοιχεία έχουν θύρες επικοινωνίας έτσι ώστε να καταφθάνουν μηνύματα απο το δίκτυο. Η αποστολή μηνυμάτων μπορεί να γίνει είτε μέσω συνδέσμων που βρίσκονται στις θύρες είτε απευθείας στο στοιχείο. Συνεπώς η

επικοινωνία ενός στοιχείου με το υπόλοιπο δίκτυο γίνεται μέσω των θυρών επικοινωνίας. Τα στοιχεία έχουν παραμέτρους οι οποίες εξειδικεύουν συγκεκριμένη συμπεριφορά σε διαφορετικό στιγμιότυπο ενός στοιχείου. Αυτές βοηθούν στην παραμετροποίηση ενός συστήματος. Τα στοιχεία στο χαμηλότερο επίπεδο υλοποιούν συγκεκριμένη συμπεριφορά με τη βοήθεια συναρτήσεων. Αυτά είναι τα simple modules τα οποία υλοποιούνται σε C++, τα οποία είτε επεκτείνουν είτε κληρονομούν συμπεριφορά άλλων κλάσεων του προσομοιωτή. Τα στοιχεία που εμφωλεύουν υποστοιχεία λέγονται σύνθετα (compound modules). Τα σύνθετα στοιχεία δεν έχουν περιορισμό στη εμφώλευση απλών στοιχειών αναπαριστώντας έτσι την δομή ενός πραγματικού συστήματος. Τα υποστοιχεία ενός compound module συνδέονται μέσω των θυρών τους, με σκοπό την επικοινωνία μέσω μηνυμάτων. Έτσι το compound module μπορεί να εκτελέσει ενέργειες μέσω της συνεργασίας των υποστοιχείων του. Η δομή των μοντέλων-στοιχειών περιγράφεται με τη βοήθεια της NED γλώσσας. Η γλώσσα NED (Network Description) υλοποιεί την περιγραφή ενός δικτύου σε στοιχεία και αποτελείται απο ποικίλες δικτυακές οντότητες (κανάλια, απλοί/σύνθετοι τύποι στοιχείων). Κάθε στοιχείο είναι ένα στιγμιότυπο ενός τύπου στοιχείου το οποίο μπορεί να αποτελέσει ένα μέρος για ένα πιο σύνθετο τύπο στοιχείου. Ανεξάρτητα στοιχεία (compound/simple) επίσης συνδέονται μέσω θυρών. Οι σύνδεσμοι έχουν παραμέτρους διακριτοποιώντας διαφορετικούς τύπους συνδέσεων, προσδιορίζοντας διαφορετική καθυστέρηση και ρυθμούς δεδομένων σε αυτές

5.2 Ipv6Suite Simulation Framework

Όπως αναφέραμε και παραπάνω, το IPv6Suite είναι μία open-source επέκταση για τον προσομοιωτή Omnet++ η οποία επιτρέπει την προσομοίωση IPv6 πρωτόκολλων και δικτύων. Ουσιαστικά επεκτείνει το INETFramework με την προσθήκη των παρακάτω RFCs:

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

αλλά και των παρακάτω Internet Drafts

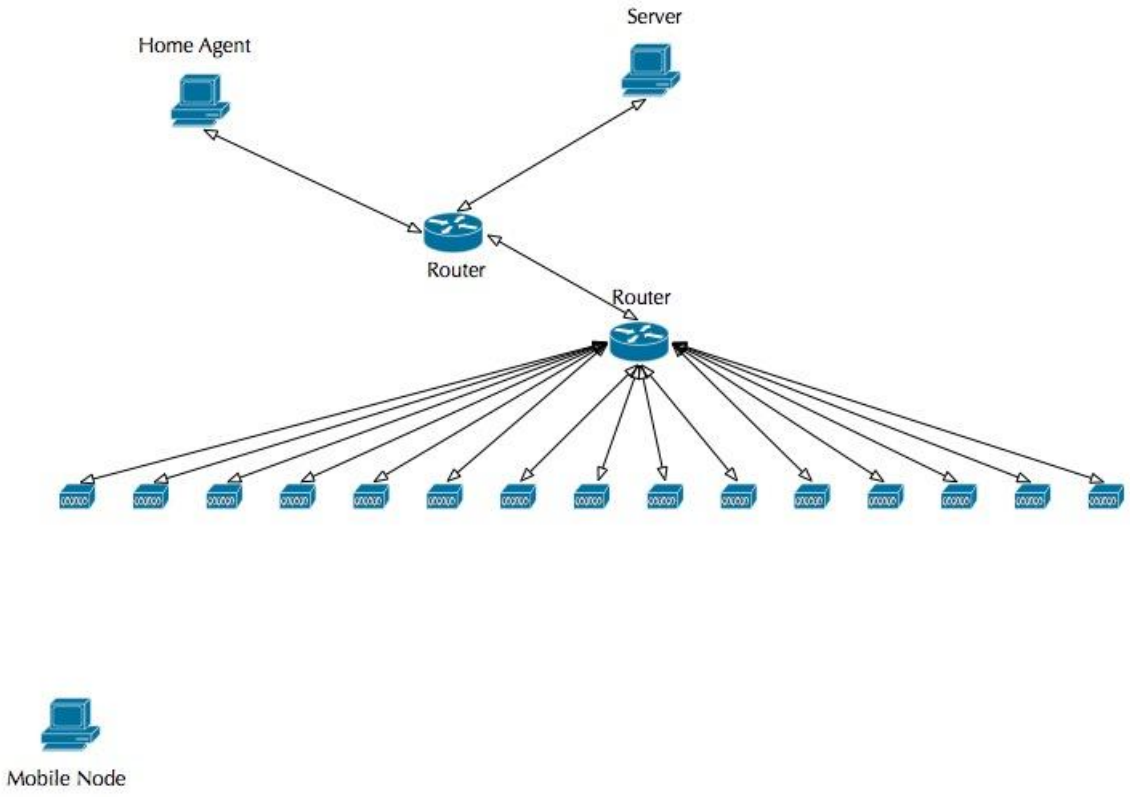
- Hierarchical Mobile IPv6 Mobility Management (HMIPv6), revision 2
- Optimistic Duplicate Address Detection, revision 0
- Fast Solicited Router Advertisements, revision 4
- Early Binding Updates for Mobile IPv6, revision 0
- Neighbor Discovery for IP version 6 - for Fast Router Solicitations

5.3 Το μοντέλο προσομοίωσης

Για την αξιολόγηση του L2 handover έναντι του L3, ως μετρήσιμα μεγέθη ορίσαμε το handover latency (την διάρκεια δηλαδή του handover) καθώς και το packet loss, τον αριθμό των πακέτων δηλαδή που χάνονται λόγω του handover.

Το μοντέλο το οποίο προσομοιώθηκε στην παρούσα εργασία, απαρτίζεται από έναν κινητό σταθμό ο οποίος κινείται σε 15 διαφορετικά foreign agents. Παράλληλα, υπάρχει ένας σταθμός στον οποίο ο mobile node αποστέλλει ICMP πακέτα ανά 10msec. Βάσει των πακέτων που λαμβάνει ο σταθμός μπορούμε να υπολογίσουμε το πλήθος των χαμένων πακέτων, αφού τα πακέτα τα οποία στέλνονται είναι εκ των προτέρων αριθμημένα. Όσον αφορά το handover latency, αυτός είναι ο χρόνος κατά τον οποίο ο τερματικός σταθμός δεν λαμβάνει πακέτα από τον κινητό σταθμό.

Στο παρακάτω σχήμα, περιγράφεται το μοντέλο μας. Όπως αναφέραμε και προηγουμένως, θα εξετάσουμε το handover της κανονικής λειτουργίας του MobileIPv6 όπως αυτή έχει προταθεί από στο RFC3775 σε αντιπαράθεση με το handover που ορίζεται από L2 triggers.



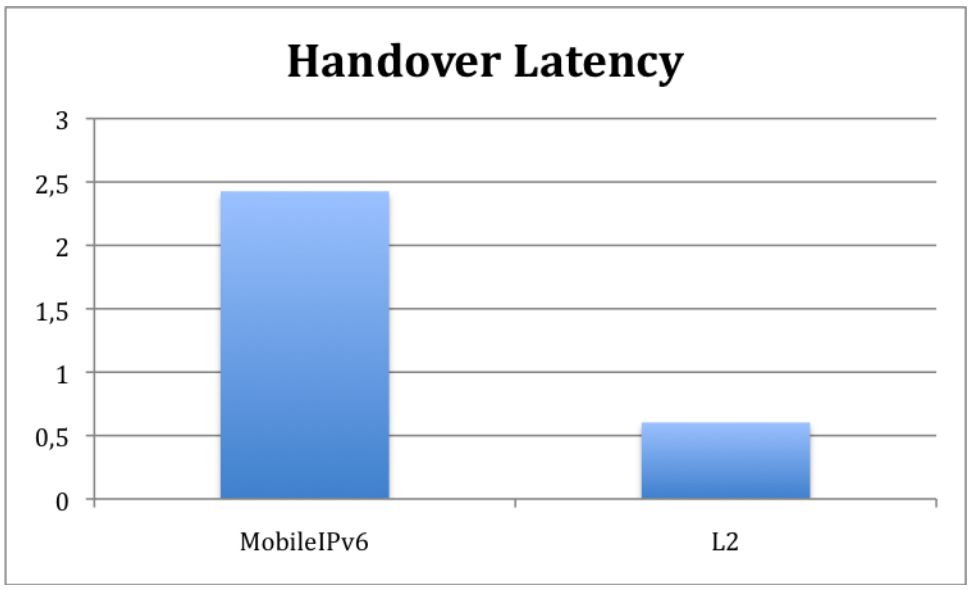
Σχήμα 35

5.4 Εξαγωγή Αποτελεσμάτων

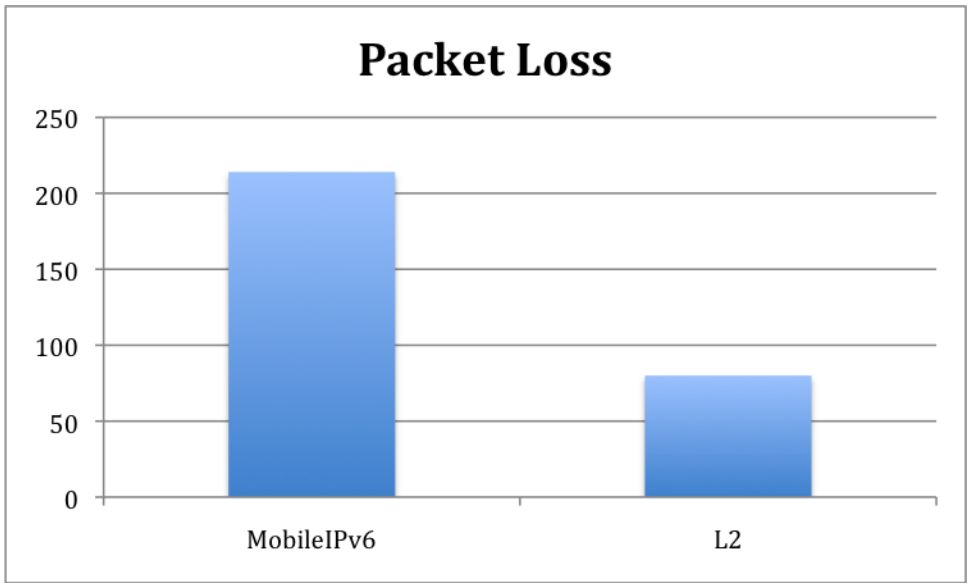
Αφού υλοποιήσαμε το παραπάνω μοντέλο, προχωρήσαμε στην εκτέλεση του πειράματος. Ο κινητός σταθμός έχει ορισθεί να κινείται με ταχύτητα 15 m/sec και η απόσταση μεταξύ του κάθε foreign agent είναι 150m. Αξίζει να σημεώσουμε εδώ ότι το κάθε πείραμα εκτελέστηκε 20 φορές για να είμαστε σε θέση να εξάγουμε ασφαλή συμπεράσματα. Βάσει αυτών των δεδομένων λοιπόν, τα αποτελέσματα συνοψίζονται στον παρακάτω πίνακα όπου αναφέρονται τα ζητούμενα μεγέθη ανά handover που πραγματοποιήθηκε.

Handover Technique	Handover Latency (msec)			Number of Packet Loss		
	Min	Max	Average	Min	Max	Average
MobileIPv6	2,18476	2,89792	**2,4258**	193	256	**214**
L2	0,48958	0,70046	**0,6033**	65	93	**80**

Σχήμα 36



Σχήμα 37



Σχήμα 38

Η απόδοση του L2 handover είναι εμφανέστατη. Η ενημέρωση του Layer 3 για ένα ενδεχόμενο handover από το Layer 2, συμβάλλει δραστικά στην μείωση του handover latency και κατά συνέπεια στο packet loss.

6 Συμπεράσματα

Στην παρούσα εργασία προσεγγίσαμε το πρόβλημα της κινητικότητας στα IP δίκτυα. Παρουσιάσαμε το Mobile IPv4 αναλύοντας τους βασικούς μηχανισμούς του όπως και τα διάφορα μειονεκτήματά του. Αναφερθήκαμε στις Route Optimization τεχνικές που έχουν προταθεί για την καλύτερη απόδοση του Mobile IP όπως επίσης και αναλύθηκε διεξοδικά και το θέμα του handover.

Το κύριο πρόβλημα του Mobile IP είναι το τριγωνικό routing, για το οποίο έχουν προταθεί διάφορες route optimization τεχνικές, uni directional και bi directional.

Αναφορικά με το handover, το κριτήριο απόδοσης ενός αλγόριθμου handover είναι η επίπτωσή του στην απώλεια πακέτων, στο latency, στο φόρτο σηματοδοσίας όπως επίσης και στη διαφάνεια του προς τα ανώτερα επίπεδα του OSI πρωτόκολλου. Περιγράψαμε μ-mobility και macro-mobility σενάρια και συζητήθηκαν handover προσεγγίσεις και για τα δύο σενάρια.

Πιο συγκεκριμένα, αναφερθήκαμε στου initiation αλγόριθμους που προδιαγράφει το Mobile IP, δηλαδή στους Lazy Cell και Eager Cell Switching αλγόριθμους. Αναφερθήκαμε επίσης και στους execution handover αλγόριθμους όπως Fast Handover via Simultaneous Bindings, FA Smooth Handover based on Route Optimization, Optimized Smooth handover based on Hierarchical Mobility Management, Position Leverage Smooth Handover και Multicast based Handover.

Μελετώντας κανείς τις εν λόγω προσεγγίσεις, αναγνωρίζει τρεις σημαντικές τάσεις που έχουν αναπτυχθεί. Η πρώτη είναι η ιεραρχική δομή των agents το οποίο σκοπό έχει να ελαττώσει τα μεταδιδόμενα μηνύματα σηματοδοσίας από και προς το οικείο δίκτυο (home network). Η δεύτερη είναι η multicast προώθηση πακέτων όπως επίσης και η χρήση buffering τεχνικών έτσι ώστε να ελλατωθούν οι απώλειες πακέτων και το latency που σχετίζεται με το handover και η τρίτη είναι η χρήση L2 πληροφορίας με σκοπό πάλι να μειωθεί το packet loss και το handover latency.

Η παρούσα εργασία έχει επικεντρωθεί στο Mobile IPv4, παρόλα αυτά αξίζει να

σημειώσουμε το Mobile IPv6 παρουσιάζει μία πιο ολοκληρωμένη λύση για τη διαχείριση κινητικότητας λόγω του ότι ενσωματώνει αρκετά από τα απαιτούμενα χαρακτηριστικά κινητικότητας.

Βιβλιογραφία

[1] C. Perkins, “IP Mobility Support”, RFC2002, IETF, Oct. 1996.

[2] C. Perkins, “Mobile IP”, IEEE Communications Magazine, May 2002.

[3] P. Ferguson and D. Senie, “Network Ingress Filtering: Defeating Denial of Service Attacks which Employ IP Source Address Spooling”, RFC 2267, 1998.

[4] D. B. Johnson and C.E. Perkins, “Route Optimization in Mobile IP”, draft-ietf-mobileip-optim-05.txt, Nov. 1996.

[5] R. Vadali, J. Li, Y. Wu and G. Cao, “Agent-based Route Optimization for Mobile IP”, Proc. of IEEE 54th Vehicular Technology Conference, Oct. 2001

[6] G. Montenegro, “Reverse Tunneling for Mobile IP, revised”, RFC 2344, 2001

[7] C. Wu, A. Cheng, S. Lee, J. Ho and D. Lee, “Bi-directional Route Optimization in Mobile IP over Wireless LAN”, IEEE, 2002.

[8] H. Hartenstein, K. Jonas and R. Schmitz, “Seamless Interdomain Handoffs via Simultaneous Bindings”, Proc. European Wireless, Dresden, Germany, Sep. 2000.

[9] E. Gustafson, A. Jonsson and C. Perkins, “Mobile IPv4 Regional Registration”, draft-ietf-mobileip-reg-tunnel-02.txt

[10] C. Perkins, “Mobile IP Local Registration with Hierarchical Foreign Agents”, draft-perkins-mobileip-hierfa-00.txt

[11] C.E. Perkins and K.-Y. Wang, “Optimized Smooth Handoffs in Mobile IP”, Proc. Of IEEE Symposium on Computer and Communications, Egypt

[12] M. Ergen, S. Coleri, B. Dundar, A. Puri, J. Walrand and P. Varaiya, “Position Leverage

Smooth Handover Algorithm for Mobile IP", submitted to World Scientific June 2002

[13] C. Tan, S. Pink and K. Lye, "A Fast Handoff Scheme for Wireless Networks", Proc. of WoW-MoM'99, Seattle, Aug. 1999

[14] K. El Malki et al., "Low Latency Handoff in Mobile IPv4" draft-ietf-mobileip-lowlatency-handoffs-v4-01,

[15] H. Yokota, A. Idoue, T. Hasegawa and T. Kato, "Link Layer Assisted Mobile IP Fast Handoff Method over Wireless LAN Networks", MOBICOM'02, Atlanta, USA,Sep. 2002.

[16] A. Chany, D. Tsangy and S. Gupta, "Impacts of Handoff on TCP Performance in Mobile Wireless Computing

[17] T. Andersen and A. Lindballe, "Seamless Handoff in Mobile IPv6", Master thesis

[18] D. Johnson and C. Perkins, "Mobility Support in IPv6", draft-ietf-mobileip-ipv6-13.txt

[19] Yong Chu Eu, Borhannudin Mohd Ali, "Multicast Based and Fast Handover Sceme in Mobile IPv6 Wireless Network", IEEE International Workshop on Antenna Technology, 2005

[20] A. Varga. "Omnet++ 3.2 User Manual"

[21] J. Lai, "IPv6Suite Simulation Framework"

ΠΑΡΑΡΤΗΜΑ

Παρακάτω ακολουθούν τα αρχεία τα οποία αναπτύχθηκαν για την υλοποίηση της παραπάνω προσομοίωσης.

Handover.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];
98
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 Handover : Handover
endnetwork
```

Handover.ini

```
[General]
preload-ned-files=*.ned @../../../nedfiles.lst
network = HANDOVER
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
HANDOVER.client1.pingApp.startTime=20
HANDOVER.client1.pingApp.destAddr = "server[0]"
HANDOVER.client1.pingApp.interval = 0.01
HANDOVER.client1.pingApp.printPing = False
HANDOVER.client1.pingApp.stopTime=1610
HANDOVER.client1*.mobilityManager.mobilityHandler.moveXmlConfig=xmldoc("HANDOVE
R.xml",
"netconf/global/ObjectMovement/MovingNode[0]")
[Run 2]
sim-time-limit = 815
HANDOVER.client1.pingApp.startTime=20
HANDOVER.client1.pingApp.destAddr = "server[0]"
HANDOVER.client1.pingApp.interval = 0.01
HANDOVER.client1.pingApp.printPing = False
HANDOVER.client1.pingApp.stopTime=810
HANDOVER.client1*.mobilityManager.mobilityHandler.moveXmlConfig=xmldoc("HANDOVE
R.xml",
"netconf/global/ObjectMovement/MovingNode[1]")
[Run 3]
sim-time-limit = 550
HANDOVER.client1.pingApp.startTime=20
HANDOVER.client1.pingApp.destAddr = "server[0]"
HANDOVER.client1.pingApp.interval = 0.01
HANDOVER.client1.pingApp.printPing = False
HANDOVER.client1.pingApp.stopTime=545
HANDOVER.client1*.mobilityManager.mobilityHandler.moveXmlConfig=xmldoc("HANDOVE
R.xml",
"netconf/global/ObjectMovement/MovingNode[2]")
[Run 4]
sim-time-limit = 415
HANDOVER.client1.pingApp.startTime=20
HANDOVER.client1.pingApp.destAddr = "server[0]"
HANDOVER.client1.pingApp.interval = 0.01
HANDOVER.client1.pingApp.printPing = False
HANDOVER.client1.pingApp.stopTime=410
HANDOVER.client1*.mobilityManager.mobilityHandler.moveXmlConfig=xmldoc("HANDOVE
R.xml",
"netconf/global/ObjectMovement/MovingNode[3]")
100
[Run 5]
sim-time-limit = 335
HANDOVER.client1.pingApp.startTime=20
HANDOVER.client1.pingApp.destAddr = "server[0]"
HANDOVER.client1.pingApp.interval = 0.01
```

```
HANDOVER.client1.pingApp.printPing = False
HANDOVER.client1.pingApp.stopTime=330
HANDOVER.client1*.mobilityManager.mobilityHandler.moveXmlConfig=xmldoc("HANDOVE
R.xml",
"netconf/global/ObjectMovement/MovingNode[4]")
[Run 6]
sim-time-limit = 280
HANDOVER.client1.pingApp.startTime=20
HANDOVER.client1.pingApp.destAddr = "server[0]"
HANDOVER.client1.pingApp.interval = 0.01
HANDOVER.client1.pingApp.printPing = False
HANDOVER.client1.pingApp.stopTime=275
HANDOVER.client1*.mobilityManager.mobilityHandler.moveXmlConfig=xmldoc("HANDOVE
R.xml",
"netconf/global/ObjectMovement/MovingNode[5]")
[Run 7]
sim-time-limit = 245
HANDOVER.client1.pingApp.startTime= 20
HANDOVER.client1.pingApp.destAddr = "server[0]"
HANDOVER.client1.pingApp.interval = 0.01
HANDOVER.client1.pingApp.printPing = False
HANDOVER.client1.pingApp.stopTime=240
HANDOVER.client1*.mobilityManager.mobilityHandler.moveXmlConfig=xmldoc("HANDOVE
R.xml",
"netconf/global/ObjectMovement/MovingNode[6]")
[Run 8]
sim-time-limit = 215
HANDOVER.client1.pingApp.startTime=20
HANDOVER.client1.pingApp.destAddr = "server[0]"
HANDOVER.client1.pingApp.interval = 0.01
HANDOVER.client1.pingApp.printPing = False
HANDOVER.client1.pingApp.stopTime=210
HANDOVER.client1*.mobilityManager.mobilityHandler.moveXmlConfig=xmldoc("HANDOVE
R.xml",
"netconf/global/ObjectMovement/MovingNode[7]")
[Run 9]
sim-time-limit = 195
HANDOVER.client1.pingApp.startTime=20
HANDOVER.client1.pingApp.destAddr = "server[0]"
HANDOVER.client1.pingApp.interval = 0.01
HANDOVER.client1.pingApp.printPing = False
HANDOVER.client1.pingApp.stopTime=190
HANDOVER.client1*.mobilityManager.mobilityHandler.moveXmlConfig=xmldoc("HANDOVE
R.xml",
"netconf/global/ObjectMovement/MovingNode[8]")
[Run 10]
sim-time-limit = 175
HANDOVER.client1.pingApp.startTime=20
HANDOVER.client1.pingApp.destAddr = "server[0]"
HANDOVER.client1.pingApp.interval = 0.01
HANDOVER.client1.pingApp.printPing = False
HANDOVER.client1.pingApp.stopTime=170
HANDOVER.client1*.mobilityManager.mobilityHandler.moveXmlConfig=xmldoc("HANDOVE
R.xml",
"netconf/global/ObjectMovement/MovingNode[9]")
[Run 11]
sim-time-limit = 160
HANDOVER.client1.pingApp.startTime=20
HANDOVER.client1.pingApp.destAddr = "server[0]"
HANDOVER.client1.pingApp.interval = 0.01
HANDOVER.client1.pingApp.printPing = False
HANDOVER.client1.pingApp.stopTime=155
HANDOVER.client1*.mobilityManager.mobilityHandler.moveXmlConfig=xmldoc("HANDOVE
```

```
R.xml",
"netconf/global/ObjectMovement/MovingNode[10]")
[Run 12]
sim-time-limit = 150
HANDOVER.client1.pingApp.startTime=20
HANDOVER.client1.pingApp.destAddr = "server[0]"
HANDOVER.client1.pingApp.interval = 0.01
HANDOVER.client1.pingApp.printPing = False
HANDOVER.client1.pingApp.stopTime=145
HANDOVER.client1*.mobilityManager.mobilityHandler.moveXmlConfig=xmldoc("HANDOVE
R.xml",
"netconf/global/ObjectMovement/MovingNode[11]")
[Run 13]
sim-time-limit = 138
HANDOVER.client1.pingApp.startTime=20
HANDOVER.client1.pingApp.destAddr = "server[0]"
HANDOVER.client1.pingApp.interval = 0.01
HANDOVER.client1.pingApp.printPing = False
HANDOVER.client1.pingApp.stopTime=133
HANDOVER.client1*.mobilityManager.mobilityHandler.moveXmlConfig=xmldoc("HANDOVE
R.xml",
"netconf/global/ObjectMovement/MovingNode[12]")
[Run 14]
sim-time-limit = 130
HANDOVER.client1.pingApp.startTime=20
HANDOVER.client1.pingApp.destAddr = "server[0]"
HANDOVER.client1.pingApp.interval = 0.01
HANDOVER.client1.pingApp.printPing = False
HANDOVER.client1.pingApp.stopTime=124
HANDOVER.client1*.mobilityManager.mobilityHandler.moveXmlConfig=xmldoc("HANDOVE
R.xml",
"netconf/global/ObjectMovement/MovingNode[13]")
[Run 15]
sim-time-limit = 120
HANDOVER.client1.pingApp.startTime=20
HANDOVER.client1.pingApp.destAddr = "server[0]"
HANDOVER.client1.pingApp.interval = 0.01
HANDOVER.client1.pingApp.printPing = False
HANDOVER.client1.pingApp.stopTime=115
HANDOVER.client1*.mobilityManager.mobilityHandler.moveXmlConfig=xmldoc("HANDOVE
R.xml",
"netconf/global/ObjectMovement/MovingNode[14]")
[Run 16]
sim-time-limit = 115
HANDOVER.client1.pingApp.startTime=20
HANDOVER.client1.pingApp.destAddr = "server[0]"
HANDOVER.client1.pingApp.interval = 0.01
HANDOVER.client1.pingApp.printPing = False
HANDOVER.client1.pingApp.stopTime=110
HANDOVER.client1*.mobilityManager.mobilityHandler.moveXmlConfig=xmldoc("HANDOVE
R.xml",
"netconf/global/ObjectMovement/MovingNode[15]")
[Run 17]
sim-time-limit = 110
HANDOVER.client1.pingApp.startTime=20
HANDOVER.client1.pingApp.destAddr = "server[0]"
HANDOVER.client1.pingApp.interval = 0.01
HANDOVER.client1.pingApp.printPing = False
HANDOVER.client1.pingApp.stopTime=105
HANDOVER.client1*.mobilityManager.mobilityHandler.moveXmlConfig=xmldoc("HANDOVE
R.xml",
"netconf/global/ObjectMovement/MovingNode[16]")
[Run 18]
```

```
sim-time-limit = 105
HANDOVER.client1.pingApp.startTime=20
HANDOVER.client1.pingApp.destAddr = "server[0]"
HANDOVER.client1.pingApp.interval = 0.01
HANDOVER.client1.pingApp.printPing = False
HANDOVER.client1.pingApp.stopTime=100
HANDOVER.client1*.mobilityManager.mobilityHandler.moveXmlConfig=xmldoc("HANDOVE
R.xml",
"netconf/global/ObjectMovement/MovingNode[17]")
[Run 19]
sim-time-limit = 100
HANDOVER.client1.pingApp.startTime=20
HANDOVER.client1.pingApp.destAddr = "server[0]"
HANDOVER.client1.pingApp.interval = 0.01
HANDOVER.client1.pingApp.printPing = False
HANDOVER.client1.pingApp.stopTime=95
HANDOVER.client1*.mobilityManager.mobilityHandler.moveXmlConfig=xmldoc("HANDOVE
R.xml",
"netconf/global/ObjectMovement/MovingNode[18]")
[Run 20]
sim-time-limit = 95
HANDOVER.client1.pingApp.startTime=20
HANDOVER.client1.pingApp.destAddr = "server[0]"
HANDOVER.client1.pingApp.interval = 0.01
HANDOVER.client1.pingApp.printPing = False
HANDOVER.client1.pingApp.stopTime=90
HANDOVER.client1*.mobilityManager.mobilityHandler.moveXmlConfig=xmldoc("HANDOVE
R.xml",
"netconf/global/ObjectMovement/MovingNode[19]")
[Parameters]
*.client1.networkLayer.proc.forwarding.routingInfoDisplay = true
HANDOVER.client1.linkLayers[*].NWIName="WirelessEtherModule"
HANDOVER.ap?.ds[*].NWIName="EtherModuleAP"
HANDOVER.server.networkLayer.proc.ICMP.icmpv6Core.icmpRecordRequests = false
HANDOVER.client1.networkLayer.proc.ICMP.icmpv6Core.icmpRecordRequests = false
HANDOVER.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"
HANDOVER.*.IPv6routingFile = xmldoc("HANDOVER.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
```

Handover.xml

```
<netconf debugChannel="HMIPv6Simple.log:rcfile:MobileMove:notice">
<!--
Ping6:Statistic:HMIPv6:custom:MIPv6MissedAdv:HMIPv6:AddrResln:MIPv6:AddressTime
r: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"/>
routeEntry routeIface="ppp2" routeDestination="0/0"
routeNextHop="3018:FFFF:0:4:5f6a:a9ff:fe2c:df2f"/>
</route>
</local>
</netconf>
```