ALMA MATER STUDIORUM – UNIVERSITA' DI BOLOGNA

FACOLTA' DI SCIENZE MATEMATICHE FISICHE NATURALI

CORSO DI LAUREA IN INFORMATICA

ARCHITETTURE PER NETWORK MOBILITY

Tesi di Laurea in Architettura degli Elaboratori

Relatore:
Prof. VITTORIO GHINI

Presentata da:
DENNIS DALLA TORRE

Sessione II

Anno Accademico 2010/2011

INDICE

1. INTRODUZIONE

Dal passaggio allo standard completamente digitale GSM (2G) del 1991, si sono moltiplicati il numero di protocolli e di tipi possibili di connessioni wireless, con diversi livelli di copertura, capacità di trasmissione e proprie caratteristiche tecniche.
I 4,6 miliardi di telefonini [1] possiedono generalmente sistemi di comunicazione a corto (infrarossi, bluetooth) ed a lungo raggio (wlan IEEE802.11) e la richiesta di accesso alla rete è continuativa ed in incremento.
Il numero crescente di utenti di Internet, lo sviluppo di nuove applicazioni basate sul protocollo IP, la commercializzazione di nuovi prodotti a basso costo, piccoli e con la necessità di connessione ovunque l'utente si trovi hanno generato l'esigenza di disporre una rete di accesso a banda larga il più affidabile ed estesa possibile.
La sfida che si sta portando avanti è quella di integrare nello stesso terminale la capacità di connettersi con molte fonti contemporaneamente e di riuscire a farlo con quelle che più si adattano alle richieste delle applicazioni e dell'utente.
Questo testo si propone di confrontare diversi tipi di soluzioni proposte e tuttora attive che hanno come obiettivo comune la gestione del traffico tra il router d'accesso alla rete ed un nodo mobile, cercando di ridurre la latenza ed il numero di pacchetti persi.
L'obiettivo di questa tesi è quello di fare il punto del livello di sviluppo e di conoscenza finora raggiunto da alcune tecnologie che hanno preso piede in campo industriale e/o di ricerca in ambito universitario. Nel capitolo due quindi cercheremo di capire quali sono le problematiche più importanti affrontate e alcune delle definizioni che ci permetteranno poi nei capitoli successivi di comprendere meglio le scelte prese dai progettisti di reti.
Nel terzo capitolo saranno spiegate le proposte delle varie scuole di pensiero per la costruzione di un'architettura mobile, cercando di valutare le caratteristiche

specifiche di ognuna che ne costituiscono i loro punti di forza o di debolezza.
Nel quarto capitolo riporteremo i risultati di emulazioni fatte in ambito universitario, dove vengono utilizzate alcune delle diverse tecnologie trattate, ed infine nel quinto capitolo commenteremo le loro conclusioni.
Vedremo dunque di capire quali sono le problematiche più comuni nell'affrontare queste situazioni e come sono state risolte tramite diverse tecnologie.

2. PROBLEMATICHE

A seconda dell'approccio che è stato dato ad ogni diverso protocollo trattato, si sono creati problemi ed opportunità diversi, sia a livello software, sia a livello hardware. Ad esempio, utilizzando più interfacce contemporaneamente nei nodi mobili si viene ad avere un consumo energetico che ne riduce anche drasticamente le prestazioni e la lunghezza di vita. Nei successivi paragrafi analizzeremo le voci più comuni e su cui sono stati fatti più lavori dai ricercatori.

2.1 Handover

Il Mobility Management è una delle funzioni più importanti di una rete GSM o UMTS e serve a gestire le procedure di ricerca, localizzazione, ricezione ed autenticazione di un telefono cellulare.
Un MM corretto permette di avere passaggi tra diversi punti di accesso della rete (vedremo in seguito le varie tipologie esistenti i handover) assicurando che non ci sia nessuna interruzione visibile dall'utente.
Più approfonditamente, gli scopi da raggiungere sono:
_ Stabilire correttamente la connessione con il punto d'accesso: il nodo mobile deve essere capace di connettersi al PoA (Point of Access) ed avere i requisiti per essere accettato.
_ Continuità di servizio: dopo l'handover, le applicazioni devono poter continuare ad essere eseguite senza alcun reset della connessione.
_ L'handover deve essere veloce e sicuro (perdere meno dati possibili).

Sono state fatte molte proposte riguardo all'implementazione del vertical handover (il cambio di tipologia di rete senza necessità di resettare la

comunicazione), che esploreremo più avanti in questo documento. E' possibile trovare riferimenti a queste tecnologie anche come “seamless vertical handover”, cioè handover caratterizzato da perdite basse o nulle di dati e valutate in base alla latenza di esecuzione, in modo da avere un efficiente passaggio tra le reti.

Il passaggio tra connessioni dello stesso tipo sono detti intra-technology o handover orizzontali, mentre i salti tra diverse tecnologie di accesso alla rete sono definiti inter-technology o handover verticali. La loro esecuzione può essere generalizzata in tre fasi:

_ Inizializzazione: Viene stabilita la necessità di un cambio di network da parte del terminale. Questo può avvenire per varie ragioni, ad esempio l'utente può aver bisogno di diversi requisiti della rete oppure viene scoperto un nuovo network più performante. Vengono quindi raccolte informazioni riguardo alle richieste QoS (Quality of Service), le preferenze dell'utente, i vincoli e le politiche dell'operatore ed i network disponibili. L'output di questa sezione è la selezione della nuova rete. Più criteri, vincoli e segnali ci sono, più il problema di scelta è complesso, anche considerato il costo in termini di tempo di preparazione, handshaking, consumi di energia e possibili perdite di dati. Decidere troppo presto di effettuare l'handover potrebbe portare a conseguenze non ottimali a causa di previsioni inesatte e mancanza di informazioni stabili (per esempio a causa di deterioramento del segnale o di disponibilità di altri network). Potrebbe addirittura venire cancellato prima del suo completamento o poco dopo a causa di problemi (per esempio l'effetto pingpong di cui scriverò in seguito) . Decisioni prese troppo tardi potrebbero al contrario far diventare inefficiente la preparazione dell'handover, causando lentezza e perdita di informazioni. Devono essere quindi scelte effettuate quando è necessario e con i giusti tempi.

_Preparazione : Questa fase prepara per il cambio di connessione e cerca di fare in modo che le azioni che verranno effettuate abbiano il minor impatto possibile sulle performance delle applicazioni in esecuzione. La gestione dipende dal protocollo di mobility management coinvolto nell'handover (che dipende dal tipo

stesso di handover) che si occupa di predisporre le risorse e, se implementato, il cambio di contesto (informazioni per quanto riguarda l'autenticazione, il profilo QoS, l'inoltro dei dati sia al vecchio sia al nuovo network).

_Esecuzione: Include gli eventi a partire dall'inizio della disconnessione dal vecchio network fino alla connessione con il nuovo. Si occupa quindi anche di inoltrare i dati mandati sia alla vecchia connessione (fino a quando il passaggio è finito), sia alla nuova, di aggiornare l'indirizzo IP e rilasciare le risorse che erano occupate dal vecchio collegamento.

Una stretta cooperazione tra le funzioni ed i protocolli che attraversano i vari layers coinvolti negli strati fisico, rete ed applicazione è essenziale per una realizzazione efficiente e senza perdite di dati.

2.2 Mobility Management con MIPv4 ed MIPv6

La base per il funzionamento del mobility management per l' IETF (Internet Engineering Task Force) è fornita da Mobile IPv4. Esso memorizza due indirizzi per ogni terminale mobile (MT), un indirizzo permanente globale (home address) che appartiene all' home network ed un indirizzo temporaneo locale (care of address CoA) presso il MT nel network che lo sta ospitando (di solito questo indirizzo è assegnato al router chiamato Foreign Agent).

Il traffico destinato al MT è sempre indirizzato all'home address e raccolto da un router chiamato home agent (HA). Questo router si occupa di inoltrare tramite tunnelling i pacchetti al foreign agent all'indirizzo COA creato precedentemente. Dopo la decapsulazione consequenziale all'arrivo, il foreign agent inoltra il traffico al terminale.

MIPv6 è stato concepito per cercare di ottimizzare la comunicazione tra i nodi mobili. Per esempio non c'è bisogno di un foreign agent, l'indirizzamento è

supportato dalle funzioni base del protocollo. Comunque la comunicazione diretta tra i nodi ed il MT presuppone che entrambi siano capaci di supportare le operazioni mobility-aware.

Questo tipo di schema introduce una significante latenza durante le procedure di handover, come la rivelazione dei movimenti (movement detection), nuove configurazioni per il care-of-address ed aggiornamenti della posizione. Se la latenza supera una certa soglia è ovviamente inaccettabile per le applicazioni real time, quindi sono stati proposti due protocolli per ridurre il tempo di handover di MIPv6 : Fast MIP v6 e Hierarchical MIPv6. Il primo è un miglioramento del protocollo che cerca di ridurre il ritardo coinvolgendo le comunicazioni tra il nodo precedente ed il successivo di accesso nella nuova rete. Il secondo aggiunge estensioni al MIPv6 che supportano la localizzazione del Mobility management (per esempio gestendo i piccoli movimenti all'interno di un network di accesso). HMIPv6 e FMIPv6 hanno capacità e funzioni complementari, quindi possono essere usate in parallelo per migliorare l'handover. La combinazione di questi due schemi è conosciuta come Fast Hierarchical MIPv6.

I protocolli descritti finora sono basati sui terminali mobili, nel senso che il MT ha bisogno di essere provvisto delle funzionalità necessarie per eseguire l'handover e di un location managment che gestisca gli spostamenti attraverso le sottoreti dei networks.

Recentemente c'è stato un interesse da parte dell'IETF di definire un protocollo per mobilità basato sulla localizzazione delle reti (NETLMM) che permetta ai MT di muoversi tra le sottoreti con lo stesso access network, senza richiedere cambiamenti nel protocollo IPv6.

Il protocollo specificato, chiamato Proxy MIPv6, introduce nuove funzioni sia per il primo router a cui si andrà ad agganciare il nodo mobile, sia per altri nodi speciali della rete di accesso, chiamata Local Mobility Anchor Points (LMAPs), che agisce come home agents. Il primo router esegue il mobility management per conto del nodo mobile fornendo, in collaborazione con LMAP, informazioni al

router che fanno credere al MT di essere ancora connesso al'home network (e gli permettono di mantenere lo stesso indirizzo). Dopo aver accertato il movimento del MT, il primo router avvia la segnalazione tramite il LMAP per aggiornare l'indirizzamento da e per il MT home address.

2.3 Effetto Ping Pong

Nelle reti mobili, un fenomeno molto comune di peggioramento delle prestazioni del network è causato dall'effetto pingpong. Concretamente consiste nel passaggio frequente tra le stesse coppie di celle del nodo mobile oppure all'alto numero di fluttuazioni di segnale tra i confini delle celle stesse. Questo incrementa il tempo di handover nella rete e pone dei sovraccarichi che il fornitore di rete stesso dovrà gestire. Secondo [8] si possono definire due tipi di effetto pingpong:

_ Traditional pingpong: quando un nodo mobile va in un'altra cella ed immediatamente dopo torna in quella originale, causando aggiornamenti negli indirizzi dei vari AP ed handover in un periodo di tempo breve.

_Generalized pingpong: quando il nodo mobile si muove in maniera circolare dentro tre o più celle, causando aggiornamenti consecutivi e ripetitivi nella gestione del suo traffico in maniera più difficilmente individuabile.

2.4 Cambiamenti infrastrutturali

Possiamo citare lavori in condizioni estreme, come la connettività in treni ad altissima velocità [2], per cui la rete va completamente ridisegnata al di là delle

possibili architetture client. E' preferibile usare una copertura lineare dell'area, anziché circolare. In Francia c'è un sistema satellite-to-Wi-Fi sui treni TGV, ma questo supporta fino a 50 passeggeri. In Inghilterra l'alta velocità fa affidamento su una combinazione di satelliti e collegamenti cellulari. La connessione satellitare purtroppo però non è l'ideale per queste situazioni, in quanto di banda limitata e con un lungo tempo di latenza (che evidentemente non lo rende ideale per applicazioni real-time e con grandi richieste di bandwidth)

3. TECNOLOGIE PER MOBILITY NETWORK

Sono state prese in esame le tecnologie che sembrano più promettenti e con più possibilità di sviluppo, sia a livello universitario che a livello industriale. Molti campi e soluzioni ai problemi sopra citati sono ancora in fase di studio e di progettazione.

3.1 Nemo

Nemo (Network MObility) è uno schema che supporta la gestione della mobilità per reti wireless mediante l'aggregazione dell'intero flusso del traffico dentro ad un singolo flusso diretto ad un Mobile Router (MR) usando lo standard MIPv6. L'intero schema è chiamato Nemo Basic Supporting Protocol (BSP).
Nemo è considerato un'estensione del protocollo Mobile IP che permette la rintracciabilità di client non fissi mediante l'utilizzo del MR. Ogni rete connessa ha un prefisso (Mobile Network Prefix MNP) ed ognuna di queste reti contiene Mobile Networks Nodes (MNN). Quando il MR si muove, usa MIPv6 per connettersi agli HA(Home Agent, router di un nodo mobile che mantiene informazioni riguardo il dispositivo corrente[4]) e ai CN (Current Networks). Nemo è più semplice di MIPv6 perchè non usa Route Optimization (RO, parte del protocollo che serve ad autenticare in modo sicuro nodi mobili senza credenziali preconfigurate o di una infrastruttura a chiave pubblica, anche in presenza di attaccanti nel percorso del nodo della vittima[5]). MR non permette servizi temporanei, ma può essere multihomed (tecnica per incrementare l'affidabilità di una connessione per IP network[6]) su internet attraverso interfacce multiple ed usare la tecnica dual-stack.
Nemo crea tunnel bidirezionali verso l'home agent esattamente come il MN per i

MNP interni: qualsiasi richiesta provenga da indirizzi esterni deve venire negata ed il MR non deve mandare pacchetti originati dalla rete mobile da nessun'altra parte eccetto ovviamente attraverso il tunnel per l'HA.

3.1.1 Come funziona Nemo

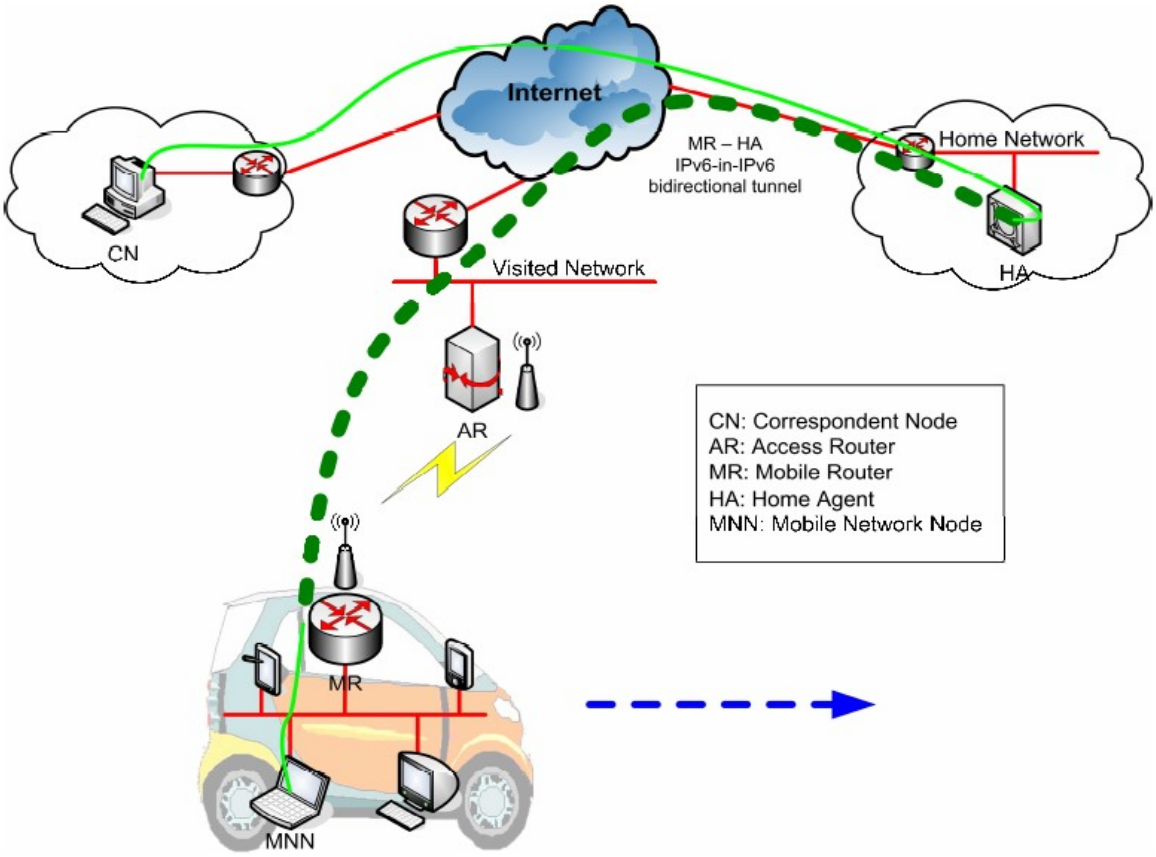


Figura [Nemo 1]

Nemo permette la mobilità delle reti usando un indirizzo IP aggiuntivo, il Care of Address (CoA), per il Mobile Router (MR). Il CoA può essere immaginato come un indirizzo temporaneo usato dal MR quando è in movimento, visto che il

cambiamento di rete lo porterebbe ad essere topologicamente sbagliato (perchè non corrispondente alla rete da cui si aggancia).
Il CoA quindi permette l'instradamento dei pacchetti dalla locazione corrente del MR, fungendo da guida. Nel frattempo il MR mantiene un altro indirizzo IP disponibile via DNS, l' Home Address (HoA), che indica la rete base (l'IP di sottorete a cui HoA fa riferimento) ed è usato per mantenere lo stato della sessione del nodo corrispondente (CNs). L'HoA agisce come un identificatore ed è usato dallo strato trasporto. Quando il MR non è nella sua rete locale, l'HA del MR (HAMR) agisce come un proxy, inoltrando i pacchetti ricevuti alla rete locale (usata dall'HoA) al MR usando il (CoA) in maniera bidirezionale, tramite IP in IP tunnel [7].
Il traffico interno alla rete mobile è mandato al MR ed incapsulato per il tunnel fino all' HA, dov'è decapsulato ed inoltrato. Questo approccio permette all'MR ed al suo nodo mobile ospite (Visiting Mobile Nodes VMN) di mantenere una connettività pseudo end-to-end nonostante il punto di connessione alla rete cambi.

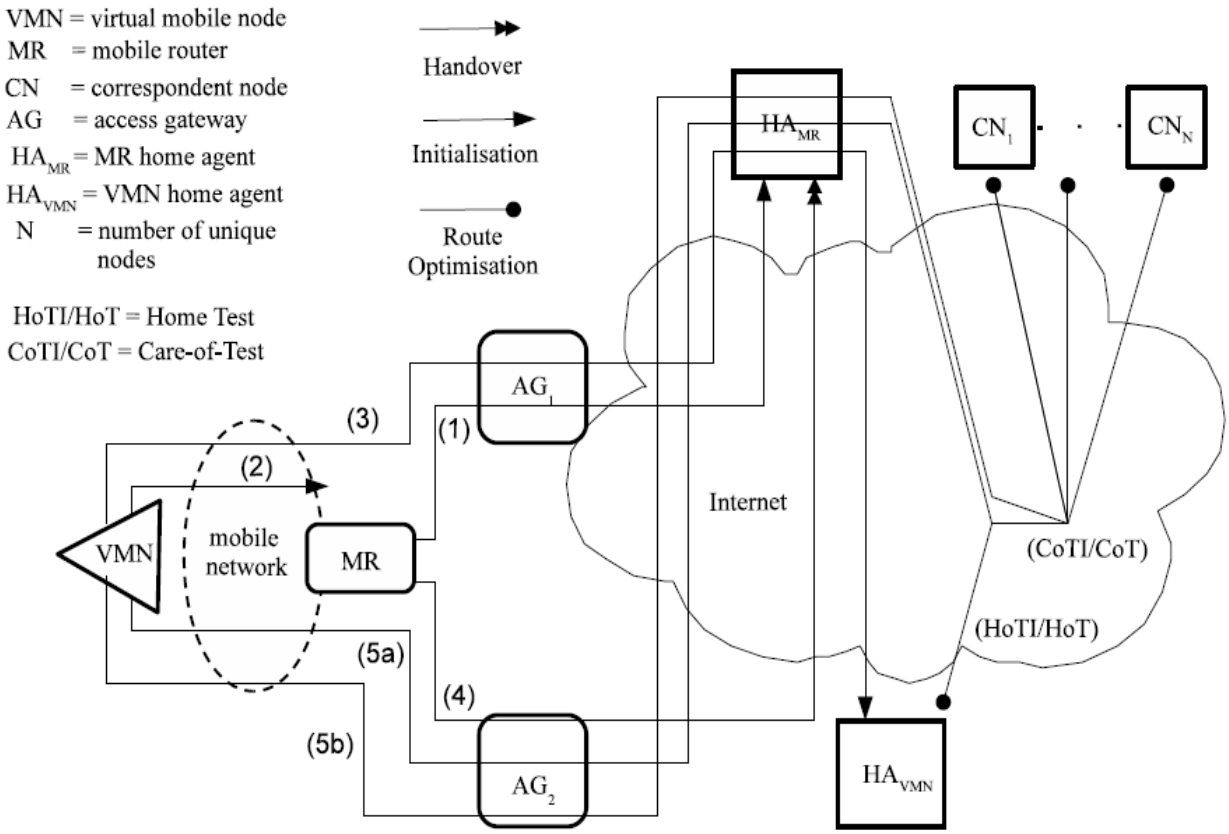


Figura [Nemo 2]: Inizializzazione ed handover di un Visiting Mobile Node (con MIPv6) e MR (con Nemo).

(1) mostra il MR aggiornare il suo HAMR (Home Address Mobile Router) tramite l'Access Gateway 1 (AG1).

(2) mostra un VMN arrivare alla rete mobile e registrare un indirizzo IP dato dal MR.

(3) mostra il VMN aggiornare il proprio HA_{VMN} con il nuovo CoA.

(4) mostra il MR muoversi e svolgere un handover informando il proprio HA_{VMN} del nuovo CoA.

(5) mostra il VMN eseguire un RRT con il proprio CNs.

(5b) mostra il VMN aggiornare il proprio CNs con un nuovo CoA.

Il VMN è capace di raggiungere questi risultati tenendo il proprio HA aggiornato con il nuovo indirizzo CoA, tramite Ipv6.

Un beneficio di questo approccio è che non cambia il modo in cui l'indirizzo IP è usato oggi, quindi non ci sono nuovi carichi o modifiche da effettuare sulle modalità o sull'architettura del protocollo.
Quando un MR con Nemo cambia punto di accesso rispetto a quello originale, risponde ad ogni routing advertisements dell'Access Gateway (AG) in modo da ricevere un nuovo CoA della rete che sta visitando. Il MR quindi manda un Binding Update (BU) al suo HAMR informandolo del suo cambio di CoA. L'HAMR aggiorna la sua tabella cache HoA-to-CoA per quel MR e risponde con un Binding Acknowledgement (BA). Queste azioni settano e mantengono la bidirezionalità dei tunnel tra loro.
I pacchetti per il MR sono ricevuti dall'HAMR, che usa l'incapsulamento del tunnel IP in IP per inoltrarli all'ultimo indirizzo CoA del MR. Tutti i pacchetti usciti dalla rete mobile (mandati da ogni VMN al Current Node) devono seguire lo stesso percorso di ritorno attraverso il MR-HAMR tunnel prima di arrivare al rispettivo proprio HA VMN(s).
L'indirizzo del MN è sempre restituito mediante una ricerca DNS. Quando questo nodo diventa un VMN e si unisce ad una rete Nemo, prima riceva il suo nuovo CoA, poi aggiorna i suoi HA VMN con il CoA mandando un messaggio Binding Update. L'HA VMN quindi risponde con un Binding Acknowledgement (BA).
Se il VMN sta comunicando con qualche mIPv6 CN, provvederà a fare un return routability test (RRT) e di conseguenza ad aggiornare i suoi CNs con il nuovo CoA tramite BU/BA (ed a mantenerli sempre tramite tunnel bidirezionali).
Operativamente, i VMN-to-HA VMN tunnel esistono attraverso il MR-to-HAMR tunnel. La mobilità del MR e del VMN è nascosta così come tutto il traffico eventualmente mandato da/a il loro rispettivo HA VMNs.
Se il MR cambia posizione, rinegozierà e riceverà il suo nuovo CoA per aggiornare l'HAMR con il nuovo posizionamento. L'HAMR quindi aggiorna la sua Binding Cache ed il tunnel bidirezionale è mantenuto. Per quanto riguarda il VMN interno della rete mobile, esso sarà all'oscuro della propria mobilità, vi sto

che il MR assicura che l'indirizzo di ingresso alla sua interfaccia rimanga immutato. La mobilità del MR colpisce solo l'interfaccia di uscita. Come risultato, il VMN non eseguirà passaggi con il suo HA VMN o il CN (se presenti).

3.1.2 Nemo ed i Firewall

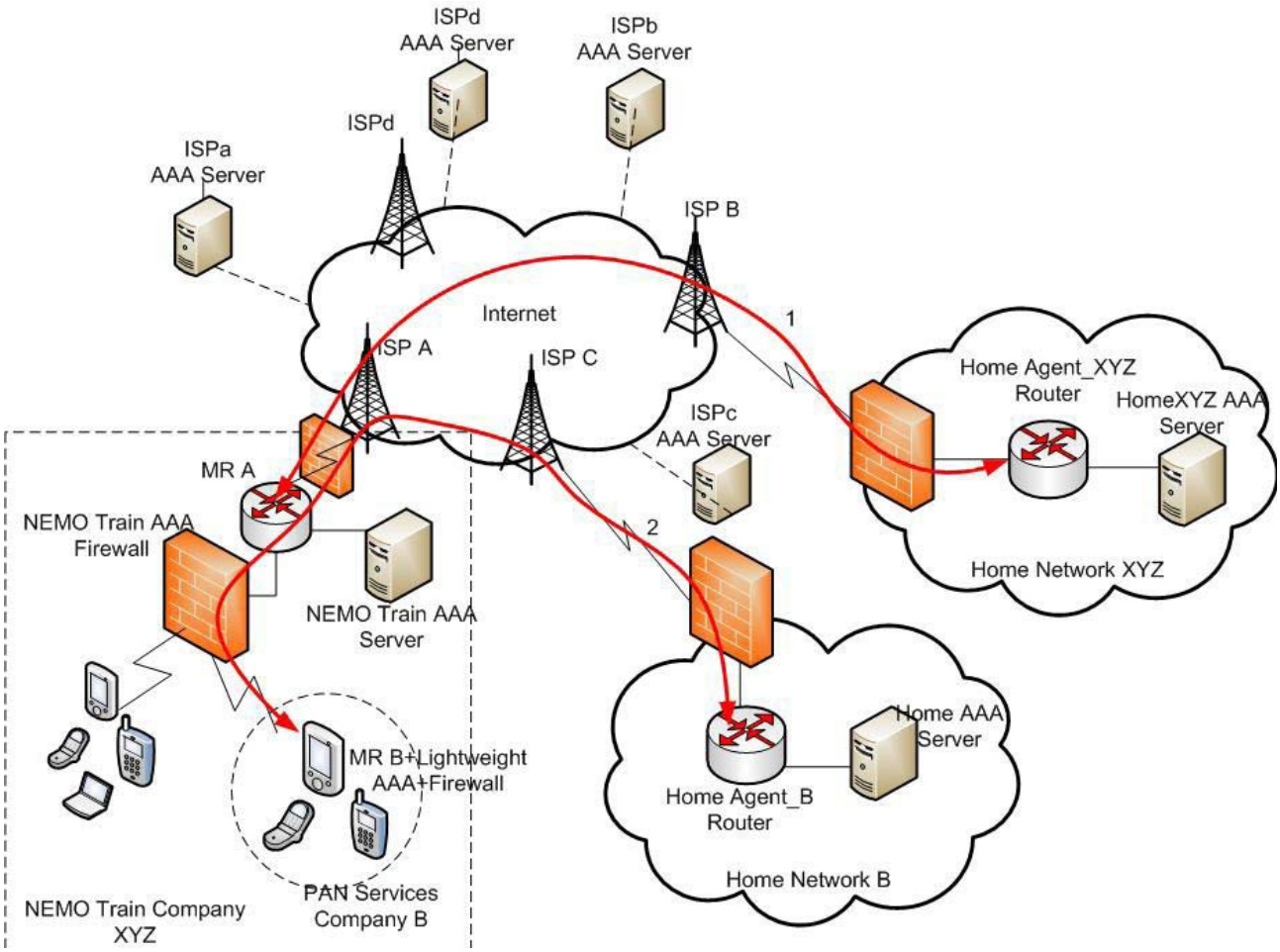


Figura [Nemo 3]

Nemo è una estensione del protocollo Mobile IPv6 creato apposta per la mobilità tra reti. Abbiamo visto nel punto precedente come funzionano i meccanismi tra

MR, HA ed il CoA. Per essere funzionale e attivo anche commercialmente, bisogna che lo si studi in ambienti dove diversi ISP (Internet service providers) lavorano in diverse posizioni geografiche. I firewalls sono comunemente e largamente posizionati negli ISP e nelle reti aziendali come importante difesa ed ovviamente, non sono stati predisposti a supportare Nemo. [11] propongono una soluzione per attraversare i firewall senza influenzare o far decadere le prestazioni ed il funzionamento del protocollo integrandolo con un server AAA (Authentication, Authorization and Accounting). I problemi fondamentali sono due: permettere al visitatore della rete di mandare via tunnel tutto il suo traffico all'HA impedendo di sfruttare questa possibilità per causare attacchi o usi inappropriati ed i firewall. Il tutto possibilmente senza dover implementare modifiche agli attuali protocolli o al software firewall. [12] propone uno schema trasversale integrato con il server AAA definendo un meccanismo di cooperazione tra il protcollo e questo tipo di sistemi di difesa.

Prima di tutto abbiamo 4 differenti ISP (Internet service provider) di 4 compagnie diverse (A B C D) [Figura 1]. Assumiamo che tutte queste compagnie stiano usando la tecnologia WiMAX per l'accesso dei MR alla rete. Gli Access Router (AR hanno il ruolo di mantenere la connettività con gli HA quando si spostano nella rete. Ogni compagnia avrà il proprio server AAA ed il proprio firewall per consentire o evitare il passaggio dei pacchetti dalla rete o alla rete.

Ipotizziamo che ci sia un treno della compagnia XYZ con il MR A che usa tecnologia Nemo a fare da tramite ai suoi passeggeri per l'accesso alla rete. Questo treno deve sempre mantenere connettività con l'HA XYZ nell' Home Network XYZ. Lì esiste un altro server AAA, chiamato XYZ_AAA. Entrambi, HA_XYZ e XYZ_AAA sono protetti da firewall di ISP B. Nel treno c'è anche un altro AAA server. I servers in questo contesto sono utilizzati per fornire cooperazione in riferimento all'Autenticazione, l'Autorizzazione e l'Accounting di Nemo.

Assumiamo che l'Home Network XYZ è in un luogo fisso ed è sempre

autenticato e connesso alla rete attraverso l'ISP B.

Per la fase AAA gli autori suggeriscono l'utilizzo del protocollo DIAMETER [14] ed il protocollo SCTP che utilizza.

Il primo tunnel che dev'essere stabilito è tra il MR A nel treno nemo e l' HA_XYZ. In questa fase il MR A è collegato all'ISP A mentre l'HA_XYZ collegato all'ISP B. Entrambi, il MR A ed il HA_XYZ sono protetti da firewalls. Secondo il protocollo Nemo, il MR A deve mandare un messaggio di Binding Update (BU) per registrare la sua posizione corrente ed ottenere l'indirizzo CoA dall'ISP A al HA_XYZ. MR A formerà un messaggio che include informazioni riguardo all'HA, il suo indirizzo, username e password. Il MR A poi creerà un messaggio AMRR (AAA Mobile Router Request) e l'inoltrerà all'ISPa Access Router (ISPaAR). L'ISPaAR inoltrerà il messaggio all'ISPaAAA server. L'ISPaAAA determinerà se le informazioni sono corrette ed in caso positivo la sessione tra MR A ed ISPaAAA è stabilita correttamente. L'ISPaAAA esaminerà le informazioni nell'AMRR e quindi inoltrerà il messaggio all'ISPbAAA.ISPaAAA aggiungerà i propri attributi AAA alle iniziali dell'AMRR e lo inoltrerà all'ISPbAAA. Un'altra sessione AAA quindi verrà stabilita tra l'ISPaAAA e l'ISPbAAA. Successivamente l'ISPbAAA inoltrerà il messaggio all' HA_XYZ AAA e manderà un'HAR (Home Agent MIP Request) per informare l'HA del CoA corrente del MR A ed allo stesso tempo viene stabilita la sessione tra l'ISPbAAA e l' HA-XYZ. Esso quindi restituirà un Binding Acknowledgement (BA) e lo inoltrerà all' HA_XYZ AAA che di conseguenza manderà un ACR (messaggio di cooperazione, AAA Cooperation Request) all' ISPbAAA, che farà lo stesso verso l'ISPaAAA per fargli aggiornare le politiche del firewall. In questo ACR includerà le informazioni che determinano come le policies sono state cambiate. Il CoA corrente diventerà l'indirizzo IP di partenza mentre quello di destinazione sarà quello del HA_XYZ. Ovviamente il firewall proibirà ad ogni altro indirizzo IP che non è stato autenticato dal server AAA l'ingresso al network.

L'ISPbAAA quindi manderà un messaggio AFR (AAA Firewall Request) al Firewall A, che aggiornerà e risponderà affermativamente con un AFA (AAA Firewall Answer) sia all'ISPa

che all' ISPbAAA. Dopo la riccezione da parte dell'ISPaAAA, un ACA (Aaa cooperation Answer) viene inoltrato all'ISPbAAA. L'ISPbBBB farà lo stesso verso l'HA_XYZ AAA ed un AFA al Firewall B.

Dopo questo, l'HA_XYZ AAA manderà un AMRA (AAA Mobile Router Answer) al MR A che conterrà le informazioni di Binding Acknowledgement e di Authentication, Authorization and Accounting che risultano dalla risposta al messaggio AMRR. L'AMRA passerà attraverso ogni sessione C, B ed A con gli attributi AAA verificati dal corrispondente server AAA ogni volta che il messaggio lo attraversa. Se l'Authorization e l'Authentication hanno successo, MR A riceverà correttamente l'AMRA e da lì ne estrarrà le informazioni del messaggio e riceverà il BA che indica che il tunnel è stato correttamente stabilito con l'HA_XYZ.

3.2 Architettura Campo

L'architettura CAMPO [3] (Context-aware Autonomic Management of Preferred network Opportunities) è stata concepita con l'intenzione di affrontare le problematiche che intervengono in caso di collegamenti multipli e di cambio di interfaccia\connessione dinamicamente, non appena viene rilevata una migliore opportunità di collegamento disponibile.

Innanzitutto viene fatto un lavoro di classificazione degli attori e delle loro interazioni. Per esempio vengono raggruppate le entità in tre tipologie:

_Le applicazioni: che rappresentano il client in esecuzione nel terminale. Richiedono in maniera attiva la connessione per soddisfare gli obiettivi che gli

vengono imposti (scaricare un hypertesto dalla rete)

_Le interfacce: l'equipaggiamento hardware disponibile nel nodo mobile (Wi-fi, Bluetooth). Possono essere attive o inattive.

_I connettori: entità che provvedono effettivamente a fornire la connessione lavorando con le interfacce attive.

Inoltre viene creata una sintassi per indicare le relazioni tra di essi. Ad esempio la classica interazione è scritta come <N:1:1>: il selettore di interfaccia è in relazione N-a-1 ed il selettore di connettore 1 a 1. In questo caso, l'intero carico della selezione dell'interfaccia è delegato all'utente finale, che può manualmente accendere o spegnere le interfacce del suo client.

Un sistema <N:M:M> indicherebbe soluzioni responsabili dell'attivazione della corretta interfaccia tra un set di possibili alternative e la selezione del connettore più adatto per ognuno di loro.

3.2.1 Vertical and horizontal handover.

Esistono quattro tipi possibili di handover:

- Inter-orizzontali (macro-mobility), dove mantenendo la stessa interfaccia, si cambia il destinatario della connessione (per esempio spostandosi tra diversi Access Point della WLAN);
- Intra-orizzontali (micro-mobility), dove il collegamento originale è rimpiazzato da una nuova connessione di destinazione all'interno dello stesso dominio del network;
- Inter-verticali (macro-mobility), si usa per indicare il cambio sia di interfaccia sia Access Point;
- Intra-verticali (micro-mobility), indica che pur mantenendo il collegamento allo stesso network, si sta cambiando l'interfaccia di

connessione.

3.2.2 Una nuova tassonomia

Secondo questo schema ci sono tre principali argomenti nei quali si possono sviluppare le scelte cruciali:
_Il management scope comprende una serie di assunzioni e conseguenti vincoli che dipendono dall'ambiente in cui CAMPO è adottato, come l'elenco degli attori attivi (interfacce, connettori), il numero di essi che possono essere attivi contemporaneamente ed il ruolo delle componenti di supporto
_L'evaluation process è incaricato di raccogliere le informazioni del contesto in cui si va ad operare e conseguentemente offrire una stima quantitativa della convenienza di ogni canale disponibile in accordo con le richieste del sistema.
_Il continuity management si riferisce a tutti quei meccanismi, strumenti e strategie che effettivamente aggiornano e cambiano a real time i canali di comunicazione e forniscono connettività senza che l'utente possa percepire interruzioni di servizio.

Ci sono tre tipi diversi di management scope:
_Interface scope, che si occupa di raccogliere informazioni a proposito delle interfacce disponibili, di attivarle ed individuare la migliore da usare e delegare ogni altra gestione delle azioni che queste eseguono al firmware/hardware associato ad essa (praticamente un selettore di interfaccia). Può anche eventualmente scegliere il connettore da utilizzare attraverso le funzionalità embedded nell'interfaccia. Il tipo di gestione che viene affidato all'interface scope è molto importante e significativo per i futuri sviluppi di CAMPO. Nei casi più semplici, la scelta del connettore di una certa interfaccia è solitamente delegata al

comportamento dei componenti integrati di basso livello della comunicazione.
_Mobile node scope, basato su dati aggiuntivi correlati al tipo di performance dei canali e le richieste/capacità di ogni nodo client. Una delle decisioni chiave di questo tipo di architettura è l'opportunità di cercare di sfruttare una o più interfacce simultaneamente. Di solito la possibilità di avere connessioni multiple attive contemporaneamente dipende dalle funzionalità del client. Se hanno risorse limitate, semplici e problemi di consumo energetico è più comune adottare, con scelta di priorità statica, una connessione per volta.
_Environment scope, che tiene conto delle conoscenze generali dell'ambiente in esecuzione, compresa la disponibilità delle componenti di supporto con i diversi ruoli (evaluation process e continuity management) nell'ambiente di esecuzione del client e /o dell'infrastruttura. Gli strumenti che utilizza sono molto vari, dal QoS per il monitoraggio dei canali alla raccolta di informazioni generale al supporto AAA (Authentication Authorization Accounting).

L'obiettivo dell'evaluation process è di misurare quantitativamente la convenienza dell'instaurazione di un certo tipo di canale di comunicazione tramite l'utilizzo di opportune interfacce /connettori. Questo tipo di valutazione è necessaria sia quando viene configurato, sia quando viene aggiornato l'elenco di canali attivi durante l'esecuzione (per poter scegliere quale è il migliore al momento).
La definizione di valutazione dell'idoneità è complessa: Una questione primaria è quella di scegliere propriamente i parametri di input da considerare (facile misurabilità e comparabilità con le diverse interfacce-connessioni disponibili a runtime). Questi parametri possono essere statici (consumo energetico, preferenze dell'applicazione-utente) o dinamici (banda, jitter, latenza..). Più sono le risorse, più è complesso e presumibilmente preciso il monitoraggio delle informazioni.
Un altro aspetto importante a livello di esecuzione è la flessibilità, cioè la

capacità di modificare e adattare il livello stesso di esecuzione a runtime e riuscire comunque a perseguire gli obiettivi posti. Come obiettivi possiamo pensare a problemi locali, come cercare di minimizzare il consumo energetico e massimizzare il livello di produttività, oppure a livello globale, quindi bilanciare il carico della rete tra tutte le connessioni disponibili e massimizzare la produttività di un network. Soluzioni integrate sembrano inadatte ad ambienti altamente dinamici, anche se sicuramente comportano meno lavoro a livello di overhead.

Il continuty management è incaricato di ricevere l'output del processo di valutazione della situazione dei canali attivi e di decidere di conseguenza gli aggiornamenti da effettuare ed i cambi di connessione/interfaccia da eseguire a runtime senza perdita di performance. I componenti del continuity management possono essere divisi in due tipi: triggers e switchers:
I triggers possono risiedere sia nell'infrastruttura, sia nel client e sono incaricati di monitorare i canali attivi (e comandare le operazioni agli switcher, quando necessario). Possono raccogliere informazioni o attendere notifiche di avvenimenti relativi alle variazioni di contesto.
Gli switchers fanno tutte le operazioni richieste per monitorare la continuità di sessione, nonostante le modifiche del canale. La progettazione di questi switchers è uno dei punti aperti di ricerca in CAMPO. Attualmente sono divisi in tre principali classificazioni in base alla loro integrazione con l'ambiente di esecuzione, il livello di granularità e di visibilità nel client.
_Integrazione stretta/leggera: stretta, quando un nuovo AP deve essere installato come nuovo equipaggiamento all'interno di una rete; leggera quando gli AP sono distribuiti fuori da una rete, tipicamente ai confini di una rete, senza impatto sull'infrastruttura già attualmente presente. In uno scenario di stretta integrazione, gli handovers all'interno del dominio richiedono solo un limitato meccanismo di supporto, visto che la connessione appartiene allo stesso dominio amministrativo

ed è possibile supporre che ci sono caratteristiche di rete simili (prima di tutto si può pensare all'utilizzo dello stesso IP). In questo caso l'unico aggiornamento richiesto è quello della posizione del client e propagarne l'identità in modo da aggiornare il routing dei pacchetti. Al contrario, handover inter-dominio solitamente forzano a cambiare dinamicamente molte delle proprietà del network e richiedono la coordinazione dei connettori disposti in diversi domini amministrativi, probabilmente gestiti anche da diversi operatori di rete (es: bisogna far passare le credenziali del client).

_La granularità definisce l'obiettivo del processo di continuity management. Il per-node switchers sposta ogni canale attivo adottando un approccio a grana grossa e sfruttando il più comodo set interfaccia/connettore d'accordo con le richieste del nodo. Al contrario il per-channel switchers può spostare un solo canale per volta, ma è più semplice da supportare e richiede meno lavoro da parte del continuty management.

_Client visibility permette di identificare il grado di coinvolgimento del client all'interno del continuity management. Per transparent continuity solution si intende uno switcher a lato infrastruttura, senza nessun coinvolgimento diretto del client, alleggerendogli il lavoro da effettuare. Al contrario l'approccio end-to-end minimizza le richieste a lato infrastruttura, delegando al continuty management tutte le operazioni. La gran parte delle soluzioni presentate per CAMPO sono in mezzo a questi due estremi. Spesso componenti a lato infrastruttura operano transparent rerouting, mentre i nodi client sono parzialmente coinvolti nell'eseguire procedure di handover.

3.3 LISP

LISP significa Locator/Identifier Separation Protocol [13]. L'idea che sta dietro a questo progetto è quella di combinare due funzioni: il Routing Locators (RLOCs, che descrive come un dispositivo è collegato alla rete) ed il Endpoint Identifiers (EIDS, che definisce chi è il dispositivo, praticamente l'indirizzo IP). I fautori della proposta di separazione di queste funzioni pensano che separare (attraverso l'utilizzo di una differente numerazione) gli spazi EIDs e RLOCs possa portare ad una maggiore scalabilità del sistema di routing, attraverso una più ampia aggregazione dei RLOCs.
Per realizzare questa aggregazione essi devono essere allocati in modo proporzionale alla topologia della rete ("Rekhter's law[10]"). Il metodo di allocazione usato dai provider per lo spazio di indirizzamento IP è un esempio di questo schema di allocazione. Gli EIDs sono tipicamente situati lungo i confini delle organizzazioni. Dato che la struttura topologica delle reti e le organizzazioni gerarchiche sono raramente compatibili, è difficile creare una numerazione che serva efficacemente entrambi gli scopi senza imporre vincoli inaccettabili (come la richiesta di rinumerazione in caso di cambiamenti del provider) per l'utilizzo di quello spazio.
LISP, in quanto specifico esempio della separazione tra LOC e ID, si propone di dividere la posizione dall'identità in modo da facilitare l'aggregazione dello spazio RLOC, implementare un'identità persistente nello spazio EID ed in qualche caso incrementare la sicurezza e l'efficienza della network mobility.

3.3.1 Implementazione della separazione Locator/ID

Ci sono due tipi di approcci per implementare la separazione tra LOC ed ID: map and encap address rewriting:

_Map and encap: è considerato l'evoluzione del protocollo ENCAPS (Bob Hinden); Quando la sorgente manda un pacchetto ad un EID al di fuori del dominio, il router di confine mappa la destinazione EID ad un RLOC che corrisponde al punto di ingresso del dominio del destinatario (c'è quindi necessità di un sistema di mappatura EID-to-RLOC). Questa è la fase di mappatura di map and encap. Nella fase encap, il router del confine incapsula il pacchetto e setta l'indirizzo di destinazione al RLOC in base alla mappa dell'infrastruttura (se c'è, altrimenti è statica). Praticamente quindi vengono aggiunti dei nuovi header al pacchetto, l'header interno con gli indirizzi di sorgente e destinazione EIDs e l'header esterno con gli RLOCs di partenza e arrivo. Quando un pacchetto incapsulato arriva al router del confine di arrivo, viene decapsulato e mandato alla sua destinazione.
Lo schema Map and Encap non richiede modifiche all'infrastruttura di indirizzamento o agli host e può lavorare sia con IPv4 che con IPv6, mantenendo l'indirizzo di sorgente originale.
Ovviamente il principale svantaggio riguarda il peso ed il lavoro(overhead) che creano questi incapsulamenti e decapsulamenti.

_Address Rewriting:basato sullo schema di riscrittura di indirizzi di Dave Clark e Mike O'Dell nelle specifiche 8+8/GSE. Sfrutta gli indirizzi a 128 bit – IPv6, usando i primi 64 bit come RLOC e gli altri come EID. Quando un host manda un pacchetto destinato ad un altro dominio, l'indirizzo di partenza contiene il suo identificatore (spesso l'indirizzo IEEE MAC) negli ultimi 64 bit ed un valore speciale (senza significato) nel RLOC. L'indirizzo di destinazione contiene l'indirizzo completo (RLOG e EID). Appena il pacchetto arriva al router di uscita del dominio, la sorgente è completata (formando un indirizzo di 128 bit) ed il pacchetto viene instradato. All'ingresso del nuovo dominio, RLOG viene riscritto con valori non specificati per fare in modo che l'host non sappia qual è il suo RLOC. Questo in teoria faciliterebbe la riscrittura necessaria per mantenere la

congruenza tra prefisso di assegnamento e topologia della rete fisica che è richiesta per questo tipo di aggressiva rinumerazione come nelle specifiche 8+8 GSE.+

3.3.2 Il Protocollo

LISP è stato progettato in modo da essere semplice, dinamico, network-based ed implementare la separazione degli indirizzi Internet tra EIDs ed RLOCs. Visto che LISP è un protocollo map-n-encap, non richiede cambiamenti agli host o alle infrastrutture di database esistenti.

E' anche un'istanza di quello che architetturalmente viene chiamato un "jack-up", visto che lo strato Network è stato "jacked up" (sollevato) ed un nuovo strato network è stato inserito sotto di esso.

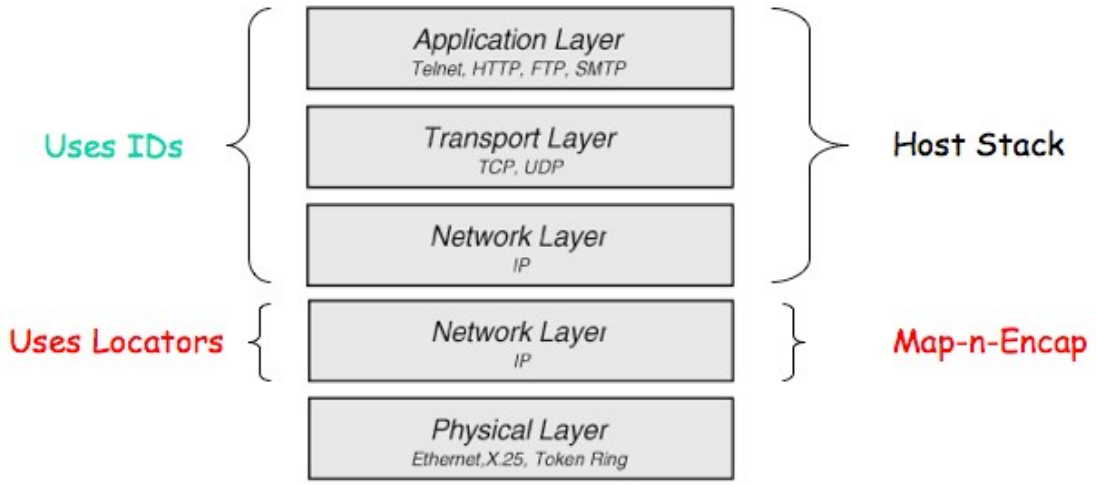


Figura [LISP 4]

LISP ha come obiettivo il miglioramento del site-multihoming (controllando per esempio l'ingresso dei luoghi senza protcolli complicati), migliorare l'ISP-

multihoming disaccoppiando l'indirizzamento del provider da quello della posizione e di ridurre la dimensione e le proprietà dinamiche delle tabelle di core routing.
Il data plane (la parte dell'architettura del router che decide cosa fare coi pacchetti nell'interfaccia in cui arrivano) ed il LISP control plane (il sistema di mappatura EID-to-RLOC) sono molto modulari.

3.3.3 LISP Network Elements

Le specifiche LISP definiscono due elementi di rete: Il router del tunnel d'uscita (ETR, Egress Tunnel Router) ed il router del tunnel d'ingresso (ITR, Ingress Tunnel Router).
Un ETR è un router che riceve pacchetti IP LISP-incapsulati dalla rete e spedisce pacchetti IP decapsulati dall'altra parte. In particolare, ETR accetta pacchetti dove l'indirizzo di destinazione nell'header risulta uno dei propri RLOCs. Il router toglie l'header letto ed inoltra il pacchetto in base al successivo header IP trovato.
Un ITR è un router che accetta pacchetti IP dall'area di cui fa da confine e li manda LISP-incapsulati attraverso la rete dall'altra parte. In particolare ITR accetta un pacchetto con un singolo header IP, tratta questo indirizzo di destinazione “interno” come un EID ed esegue una ricerca sul relativo mapping EID-to-RLOC se necessario. Il router quindi antepone un header IP esterno con un indirizzo RLOC indirizzabile globalmente nel campo dell'indirizzo di sorgente ed il risultato del mapping nel campo dell'indirizzo di destinazione. Ovviamente questo indirizzo RLOC potrebbe essere solo un proxy intermedio che ha una migliore conoscenza della geografia EID-to-RLOC più vicina alla destinazione EID.

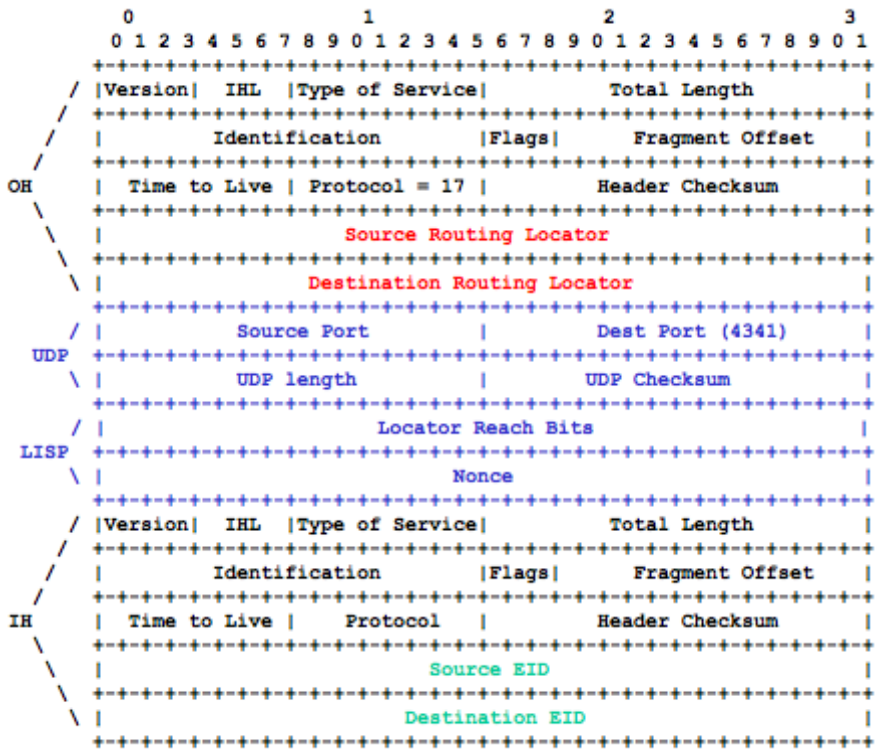


Figura [LISP 5]: Formato dell'header di un pacchetto.

3.3.4 Struttura dei dati in LISP

Quando un host nel dominio di LISP crea un pacchetto, mette il suo EID nell'indirizzo di partenza e l'EID dell'host con cui vuole comunicare nell'indirizzo di destinazione (che verrà cercato nel DNS). Se il pacchetto di destinazione è in un altro dominio, esso attraverserà l'infrastruttura della sorgente fino all' ITR. L'indirizzo EID-to-RLOC che corrisponde all'ETR è nel dominio di destinazione o nel proxy del dominio di destinazione.

Quando il pacchetto arriva all'ETR finale, viene decapsulato e inoltrato alla sua

vera destinazione.
Come menzionato sopra, le specifiche LISP definiscono tre tipi di pacchetto per supportare il sistema di mappatura EID-to-RLOC. Il primo tipo di pacchetti, il Data-Probe, è un pacchetto che un ITR può mandare al sistema di mappaggio per cercare il giusto percorso. L'apposito ETR risponderà all'ITR con un Map-Reply message notando che in questo caso l'indirizzo di destinazione interno è stato copiato al posto dell'indirizzo interno dall'ITR. (cioè il destination address interno equivale al destination address interno ed è un EID).
Il secondo tipo di pacchetti LISP è usato per supportare questo sistema di Map-Request. Un ITR può domandare la mappa mandando una richiesta Map-Request al sistema di mappaggio per richiedere uno specifico percorso. Così come nel caso del Data-Probe, il corrispondente ETR risponderà con un Map-Reply message.
Il terzo tipo di pacchetto LISP è appunto il Map-Reply. Un ETR manda un Map-Reply sotto due condizioni. La prima, se riceve un pacchetto LISP in cui l'indirizzo di destinazione dell'header interno è uguale a quello dell'header esterno (come scritto sopra), poiché questo gli fa capire che è un pacchetto Data-Probe. La seconda, se riceva una Map-Request.

3.3.5 LISP-ALT, la gestione delle mappe

L'idea base dietro LISP-ALT (o LISP Alternative Topology), è di costruire una topologia di rete alternativa per gestire i mapping EID-to-RLOC per LISP. Questa topologia logica usa esistenti tecnologie e strumenti (specialmente il BGP, Border Gateway Protocol) e le sue estensioni, insieme con il Generic Routing Encapsulation per costruire un network sovrastante di dispositivi che usano solo prefissi EID. Un importante obiettivo del design di questo LISP-ALT

è di minimizzare il numero di cambiamenti necessari all'hardware o al software esistente per deporre il sistema di mappaggio. Come tale, LISP-ALT non richiede modifiche né a BGP né a GRE.

LISP-ALT è una architettura ibrida push/pull. Quando c'è bisogno di aggiungere un prefisso EID, viene “pushato” fra i router LISP-ALT e opzionalmente agli ITR. Percorsi specifici EID-to-RLOC possono essere “pulled” dagli ITR sia tramite Map-Request o Data-Probe.

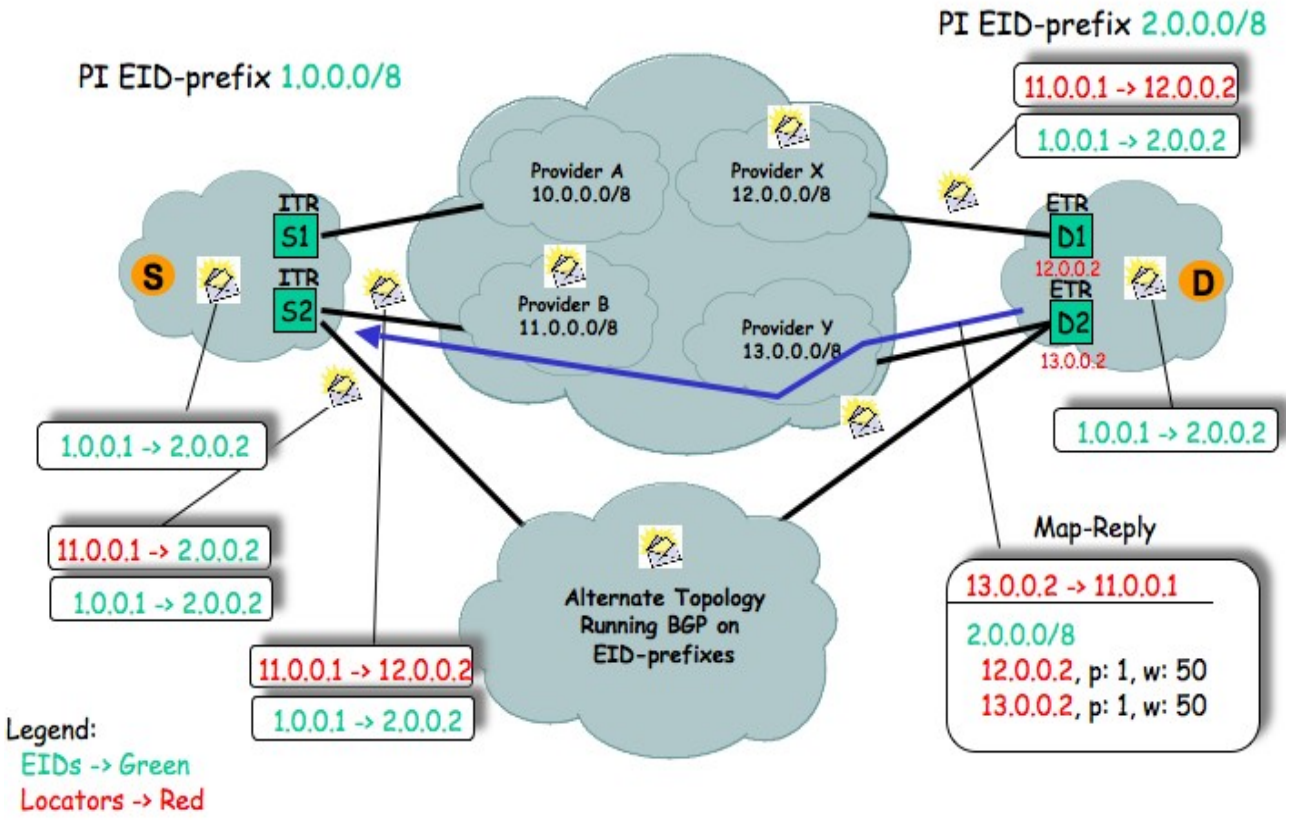


Figura [LISP 6]: schema di funzionamento pratico.

3.3.6 Esempio pratico di come funziona LISP

Quando un host LISP vuole spedire dei dati, prima di tutto cerca l'indirizzo dell'host EID corrispondente nel DNS, quindi lo mette nel campo dell'indirizzo di destinazione del pacchetto e mette il proprio nel campo dell'indirizzo di partenza. Se la destinazione è in un altro dominio, il pacchetto attraverserà il dominio fino all'opportuno ITR (Ingress tunnel router). Arrivato lì, verrà settato l'indirizzo dell'RLOC di destinazione (mappato per l'opportuno EID) in un header che andrà ad incapsulare il pacchetto e verrà messo come indirizzo di partenza l'RLOC dell'ITR. Il pacchetto è quindi mandato attraverso la rete all'ETR (Egress tunnel router) indicato e lì verrà decapsulato e mandato all'EID di destinazione.
Se l'ITR non ha la mappatura del percorso EID-to-RLOC corretta, incapsula il pacchetto con un header LISP nel cui campo di destinazione mette l'indirizzo già presente nell'header originale, cioè l'host EID. Questo è chaimato Data Probe packet, ed è instradato verso un router LISP-ALT (tipicamente un ETR, ma non è richiesto) predisposto per il mappaggio EID-to-RLOC. Quando l'ETR riceve il Data Probe packet, lo decapsula e lo inoltra all' EID di destinazione ed in risposta manda un Map-Reply alla sorgente, in modo che i pacchetti che verranno mandati in sequenza allo stesso indirizzo possono essere mandati naturalmente attraverso la rete senza dover usare di nuovo il LISP-ALT network. Questo tipo di domanda/risposta è solo richiesta quindi per il primo pacchetto tra i siti; ITR comunque può anche precaricare nella propria cache la mappa per le destinazioni più comuni usando il messaggio Map-Request, evitando pacchetti Data Probe e la latenza che ne verrebbe associata.
Per esempio, considerando la figura 3, la sorgente S con EID 1.0.0.1 vuole mandare un pacchetto alla destinazione D il cui EID è 2.0.0.2. Il pacchetto arriva all'ITR S2, che non ha una mappa per 2.0.0.2 . S2 LISP incapsua il pacchetto con l'header esterno lasciando RLOC 11.0.0.1 come indirizzo sorgente, copiando la destinazione EID 2.0.0.2 dall'header interno a quello esterno di destinazione e

mandando il pacchetto (Data Probe quindi) all'interno di LISP-ALT. Il pacchetto segue il percorso calcolato da BGP (Border Gateway Protocol) fino a ETR D2. Quando D2 lo riceve, lo decapsula e lo inoltra alla destinazione 2.0.0.2; D2 inoltre risponde con un Map-Reply message che dice a S2 11.0.0.1 che la mappa EID-to-RLOC per 2.0.0.0/8 ha due elementi, ETR D1 (il cui RLOC è 12.0.0.2) ed ETR D2 (RLOC 13.0.0.2). Dopo aver ricevuto questo messaggio, l'ITR S2 può mandare un pacchetto incapsulato LISP attraverso la rete (e non attraverso ALT). Bisogna notare che la richiesta mapping ha gli attributi priorità (p) e peso (w). La priorità serve a dire all' ITR quale ETR usare ed in che ordine, e il peso dice all' ITR come dividere il carico attraverso gli ETR di una certa priorità (w è la percentuale di traffico che deve andare attraverso ogni ETR). In questo caso, entrambi gli ETR hanno la stessa priorità (1) e lo stesso peso (50).

3.3.7 Considerazioni

Ci sono due tipi di questioni da valutare quando si cerca di capire quanto è efficiente LISP: Il lavoro addizionale causato dall'overhead e la latenza causata dalla ricerca EID-to-RLOC (oltre ai pacchetti persi).
Riguardo all'overhead, la preoccupazione è che l'aggiungere header LISP causerà un sovradimensionamento del pacchetto, eccedendo il MTU (Maximum Tansmission Unit). Questa è un'area ancora di ricerca [19].
Parlando di latenza e pacchetti persi, siccome LISP-ALT usa BGP per trovare un particolare percorso EID-to-RLOC, potrebbe esserci un certo ritardo associato ai primi pochi pacchetti mandati nel primo flusso tra i siti (i successivi beneficeranno del fatto che il percorso sarà già nella cache). Comunque, questa latenza è mitigata ed i primi pacchetti non sono persi dato che LISP può mandarli come Data-Probe. C'è anche della latenza associata al tempo richiesto all'ETR di

destinazione per rispondere con un Map-Reply. Comunque, una volta che questo è stato fatto, non c'è altro tempo da perdere per quanto riguarda la latenza indotta dal mapping.

3.4 MONAMI

Monami è un'estensione di MIPv6 la cui particolarità è permettere di registrare più indirizzi IPv6 per ogni nodo terminale di rete, più precisamente uno per ogni interfaccia. Questo a livello teorico permette di aumentare la dinamicità dell'host, il multihoming e le performance delle applicazioni che possono cercare connessioni sempre più adeguate ai bisogni dell'utente, anche contemporaneamente tramite più tecnologie.
[15] cerca di quantificare quanto migliorano le prestazioni usando questo sistema.
I nodi hanno la capacità di avere più accessi in entrata ed uscita contemporaneamente, viste le numerose tecnologie disponibili. I protocolli per la gestione della mobilità, specialmente MIPv6, sono capaci di fornire accesso ininterrotto anche in presenza di handover, anche con un certo livello di qualità. Ciononostante, hanno limitazioni per quanto riguarda il supporto di interfacce multiple, visto che è norma avere un singolo Home Address (HA oppure HoA) ed un singolo Care-of-address(CoA) accoppiati (binding pair). La registrazione di più indirizzi (MCoA, Multiple Care-of-address Registration), può superare queste limitazioni creando nuova elasticità verso i fallimenti, visto che se una connessione non va a buon fine, se ne può usare un'altra che è già stata registrata precedentemente senza perdite [16] [17]. Ovviamente questo sistema porta anche degli svantaggi, in particolare il MCoA non specifica nessun meccanismo per usare diversi CoA che sono stati registrati, per esempio un'applicazione per il

download può essere interessata ad una connessione con alte capacità di trasmissione, mentre il VoIP ha bisogno di averne una con poca latenza.
Oltre a questo, sono richiesti anche meccanismi per condividere le informazioni tra i diversi protocolli e le applicazioni e qualche programma che possa informare ed essere informato quando nuovi indirizzi sono disponibili (e quali servizi offrono).
MCoA introduce un nuovo elemento, il Binding Identifier (BID), numero che è usato per identificare le associazioni, distinte dal CoA di MIPv6. Questa estensione introduce modifiche nelle strutture dati, come la Binding Update List e la Binding cache e modifica i messaggi come Binding Update e Binding Acknowledgement. Quest'ultima modifica è necessaria per trasmettere le informazioni riguardanti i MCoA e le rispettive registrazioni.
Ci sono due modi effettuare la registrazione. Il primo è la registrazione di massa, che permette ad indirizzi multipli di essere registrati con un singolo messaggio BU. Con questa modifica esso è esteso per permettergli di includere diverse opzioni BID che specificano l'indirizzo da registrare.
Il secondo sistema consiste nell'usare un BU per ogni singolo indirizzo. Attualmente questo modo è l'unico che è supportato dal nodo corrispondente, per evitare la complessità di un'instradazione su un set di indirizzi. Ovviamente i nodi che partecipano al binding (HA e MN) devono entrambi supportare questa modalità o scambiarsi un messaggio BA per declinare l'offerta.

4. PRIMA EMULAZIONE : NODI IN MOVIMENTO

4.1 La Metropolitana di Londra

[18] mette a confronto Nemo, OptiNets (estensione di Nemo con che tenta di ottimizzare il routing), ILNPv6 (Identifier Locator Network Protocol) dal punto di vista dei costi, dei pacchetti persi e del bandwidth overhead basandosi su una emulazione fatta prendendo come scenario la metropolitana di Londra Circle Line. E' stata fatta questa scelta non in base a modelli di mobilità particolari, ma per provvedere a creare uno scenario con come input numeri realistici, piuttosto che scegliere valori arbitrari.
L'idea dietro l'esperimento è quella di portare alla luce quanto costa fornire questo servizio di ottimizzazione di instradamento (per network mobili) con i diversi approcci. Visto che l'interesse è solo nelle differenze di approccio di progettazione tra il Naming ed il Tunnelling, l'esclusione di ogni effetto wireless ci permette di tirare conclusioni con certezza basandoci solo sulle differenze dei protocolli di queste architetture e rende l'emulazione meno complessa.

4.2 Premessa

Abbiamo già visto come funziona Nemo, ora prima di presentare i risultati della ricerca, spieghiamo brevemente cosa sono OptiNets ed ILNPv6.
Con il protocollo Nemo, i VMNs all'interno di un network non sono consapevoli di star muovendosi, visto che il loro indirizzo non cambia. Questa inesattezza topologica (uno sfortunato aspetto del tunnelling) fa diventare impossibile

l'ottimizzazione della rete da parte del MIPv6 del VMN. In seguito, tutti i nodi mobili all'interno della rete hanno un corretto CoA. Questo permette al VMN con MIPv6 di eseguire Route Optimisation (RO) con un CN RO-aware, tramite un Return Routability Test (RRT) (Figura 8(b)) ed un Binding Update (BU) (Figura 10).

Un nodo mobile ha il proprio Home Address (HA_{VMN}), che è sempre trovabile tramite ricerca DNS. Quando questo nodo diventa un VMN e si unisce ad una rete OptiNets, deve prima di tutto ricevere il suo nuovo CoA (Figura 8 parte 2), quindi aggiornare il proprio HA_{VMN} mandando un BU (Figura 8 passo 3 e Figura 9a). L'HA_{VMN} quindi risponde con un BA. Se il VMN sta comunicando con un qualsiasi altro CN MIPv6, eseguirà un return routability test (RTT) (Figura 8 passo 5a e Figura 9b)e di conseguenza aggiornerà il suo CN con il nuovo CoA, via BU (Figura 8 passo 5b e Figura 10). Così ora è possibile che il suo indirizzo sia topologicamente corretto.

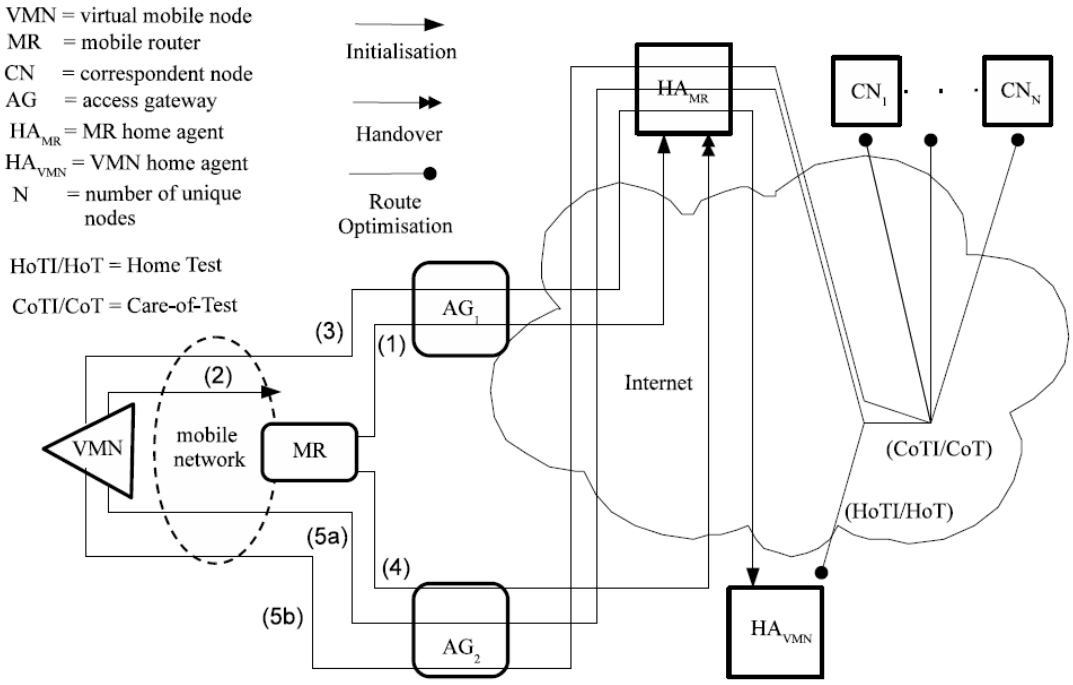


Figura [OptiNets 8]

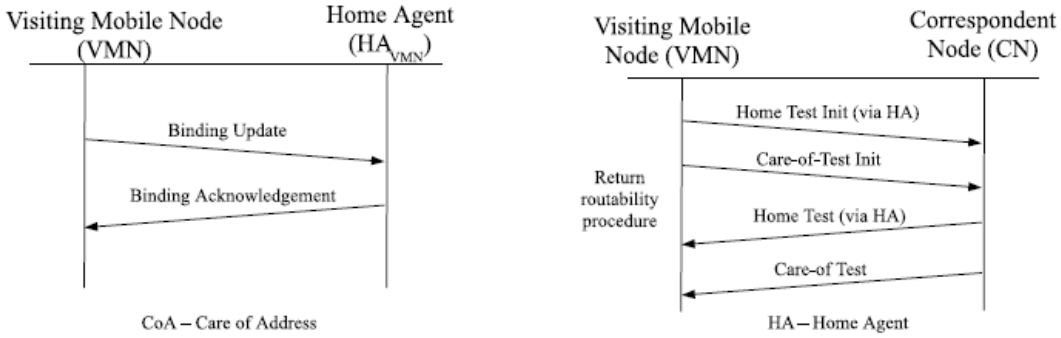


Figura [OptiNets 9a 9b]

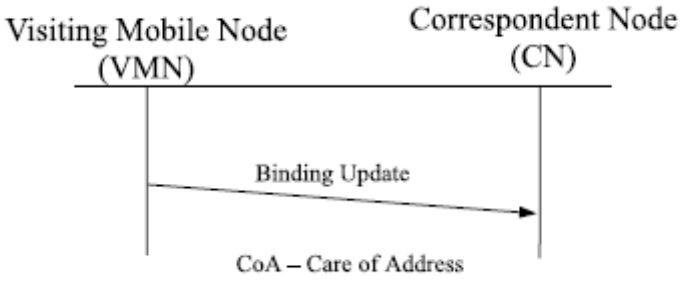


Figura [OptiNets 10]

ILNPv6 (Identifier-Locator Network Protocol for IPv6) è usato come miglioramento di IPv6. Tramite questa tecnologia, l'indirizzo finale è dinamicamente legato a due parti, una indipendente dalla topologia della rete, l'identificatore (Identifier I) ed il locatore (Locator L). Il Locator non è visibile sopra lo strato network, gli strati superiori sono legati infatti al valore dell'Identifier. In ILPNv6 il Locatore usa la stessa semantica e gli stessi bits degli indirizzi IPv6, quindi passano senza problemi attraverso lo stato network esistente che usa IPv6. L'identifier conseguentemente occupa i 64 bits inferiori dell'interfaccia, ma ha una differente semantica che identifica un nodo e non un'interfaccia.

In ILNPv6 le reti usano un indirizzamento privato interno alla rete stessa ed il MR del network riscrive i valori dei Locator appena un pacchetto ci transita dentro. La riscrittura effettuata dal Locator non comporta cambiamenti ad altri protocolli (TCP), visto che solo l'Identifier è usato dallo strato di Trasporto. I nodi che sono connessi alla rete hanno un campo DNS LP che punta ad un comune DNS L64, che sarà aggiornato dal MR quando il suo collegamento si muoverà in un'altra rete IPv6.

4.3 L'emulazione

Sono stati usati dati statistici dei treni in movimento e simulato il traffico di passeggeri per comparare le performance di NEMO, OptiNets ed ILNPv6 in uno scenario di rete mobile. Lo scenario, come già indicato, è la metropolitana circolare di Londra (London Circle Line), UK. Assumiamo che i passeggeri che salgono e scendono dai treni sono VMNs (Visitor Mobile Nodes) e consideriamo ogni treno come un network mobile distinto ed ogni arrivo del treno in una stazione come un movimento che richiede un nuovo punto di attacco alla rete. Sono stati utilizzati due set di dati riguardanti la Circle Line, raccolti da Tubeprune [20] e Transport of London [21]. I dati grezzi usati sono schematizzati nella tabella I ed i dati derivati nella tabella II.

Hours of service per day (N_d)	18
No. of trains per station per hour (N_t)	7
No. of stations per hour (N_s)	27
Mean no. of passengers on a weekday (N_w)	218136

Tabella I

No. of passengers (VMNs) per hour per train (N_p)	356
Handover/stop time at stations per train (T_h)	60s

Tabella II

Abbiamo iniziato con un uguale numero di passeggeri a bordo di ogni treno ed in ogni stazione (356). Assumiamo che questo numero rimanga costante durante l'esperimento (quindi ogni passeggero fa un solo viaggio durante quel giorno e l'inizializzazione del VMN viene fatta una volta per passeggero per treno al giorno).

Le variabili che consideriamo in questo esperimento sono il numero di stazioni che un passeggero attraversa (handovers Nh) ed il numero di CN unici per treno (N_{CN}). Per OptiNets influisce l'handover del VMN [Figura 9a] ed il numero di aggiornamenti VMN-to-CN [Figura 9b]. Ci sono anche un certo numero di pacchetti generati dall'inizializzazione e dagli handover MR-to-HA in entrambi i protocolli. Le inizializzazioni VMN sono ovviamente dipendenti dal numero di passeggeri (Np) e quello degli handover MR-to-HA dal numero di stazioni (Ns). [Np *Ns * Th] è definito come il tempo in cui tutte le registrazioni e gli handovers devono essere completati per ogni passeggero per stazione. Th è stato definito come 60 secondi (il tempo medio che un treno spende ad ogni stazione) assumendo che l'handover viene fatto alla stazione.

_Nemo: L'overhead generato da Nemo per passeggero per treno per secondo, [Cnemo] è calcolato come:

$$C_{NEMO} = \frac{K_1.N_p + K_2.N_p.N_{CN} + K_3.N_s}{N_p.N_s.T_h}$$

Figura [Metro 11]

Dove K1, K2 e K3 sono costanti.

K1*Np si riferisce all'inizializzazione VMN (VMN ed HA si scambiano BU e BA) [Figura 12a].

K2*Np*Ncn è l'ottimizzazione dell'instradamento con il CN esistente. Questo include un RRT (return routabiliy test [Figura 12c]) ed un BU/BA [Figura 12b].K3*Ns è l'overhead generato dal MR per treno per ora per l'handover. [Figura 12d].

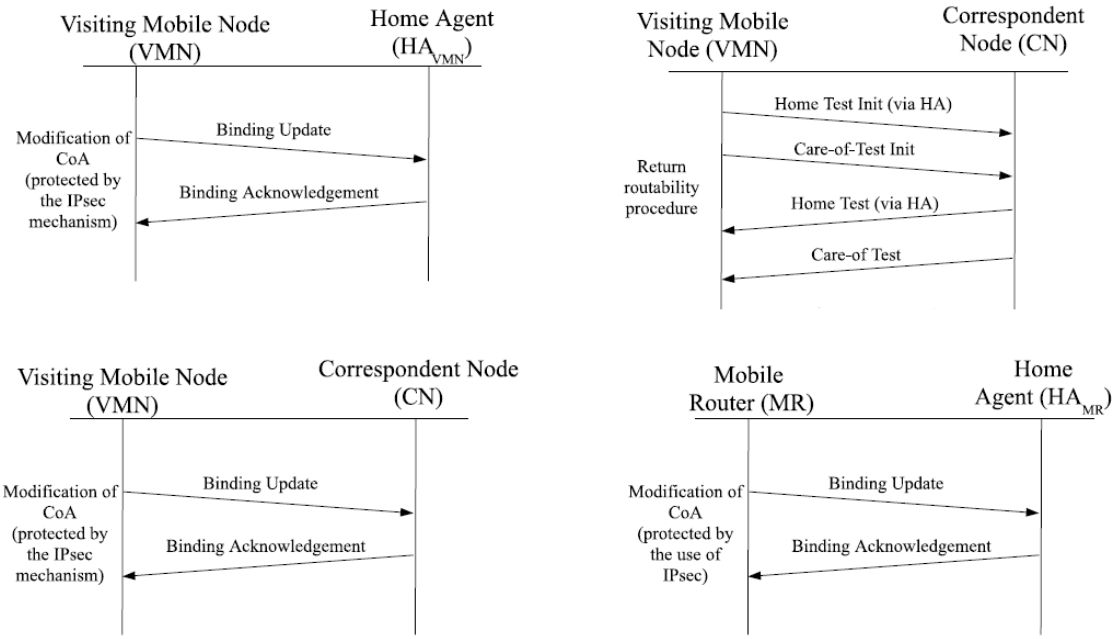


Figura [Metro 12 a b c d]

a) Inizializzazione VMN: 2 pacchetti, 284 bytes;

b) Test di raggiungibilità: 4 pacchetti, 536 bytes;

c) VMN-CN Route Optimization: 2 pacchetti, 300 bytes;

d) MR handover: 2 pacchetti, 228 bytes;

Ci sono 27 stazioni, Ns ed ogni treno ci mette circa un'ora di media a finire il suo percorso, quindi calcoliamo il numero di MR handover generato in K3*27

Se sostituiamo K1, K2 e K3 con il numero di pacchetti o di byte spediti, abbiamo rispettivamente l'overhead (Nnemo) e l'overhead di banda (Bnemo).

$$C_{NEMO} = \frac{K_1.N_p + K_2.N_p.N_{CN} + K_3.N_s}{N_p.N_s.T_h}$$

$$N_{NEMO} = \frac{2.N_p + 2.N_p.N_{CN} + 2.N_s}{N_p.N_s.T_h}$$

$$B_{NEMO} = \frac{284.N_p + (300).N_p.N_{CN} + 228.N_s}{N_p.N_s.T_h}$$

Figura [Metro 13]

_OptiNets: L'overhead generato da OptiNets per passeggero per secondo Copti è calcolato come da [Figura 14]:

$$C_{OPTI} = \frac{H_1.N_p + H_2.N_p.N_{CN}.N_h + H_3.N_s}{N_p.N_s.T_h}$$

Figura [14 Metro]

dove H1, H2 e H3 sono costanti. Esattamente come con Nemo, le tre parti dell'equazione si riferiscono all'ingresso nella rete [Figura 15a], la route optimisation con il RRT[Figura 15b] ed il BU [Figura 15c], ed infine l'overhead [Figura 15d],. Nopti è quindi l'overhead e Bopti l'overhead di banda.

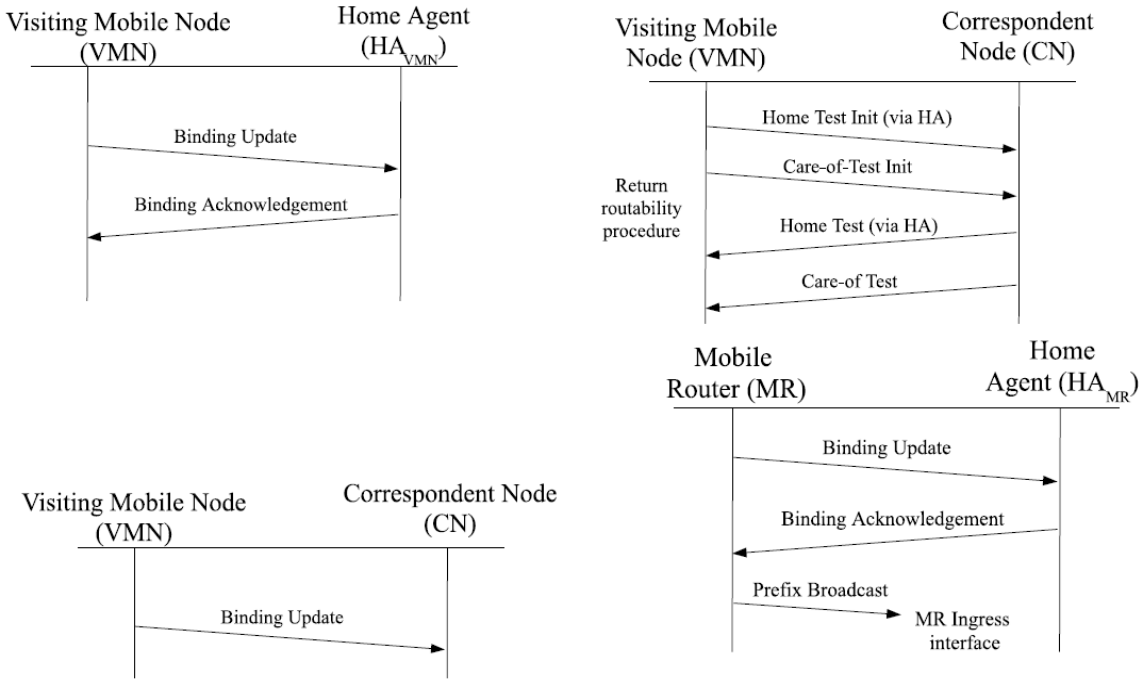


Figura [Metro 15 a b c d]

a) Inizializzazione VMN: 2 pacchetti, 284 bytes;

b) OptiNets Test di raggiungibilità: 4 pacchetti, 508 bytes;

c) Optinets VMN-CN Route Optimization: 1 pacchetti, 96 bytes;

d) Optinets MR handover: 3 pacchetti, 488 bytes;

$$N_{OPTI} = \frac{2.N_p + 5.N_p.N_{CN}.N_h + 3.N_s}{N_p.N_s.T_h}$$

$$B_{OPTI} = \frac{284.N_p + (508+96).N_p.N_{CN}.N_h + 488.N_s}{N_p.N_s.T_h}$$

Figura [16 Metro]

_ILNPv6: Sempre con alcune somiglianze rispetto ai casi precedenti, le tre parti corrispondono all'inizializzazione del VMN [Figura 19 a], quindi all'overhead generato dall'handover del MR che aggiorna la sua posizione [Figura 19b] (dipende solamente dal numero di stazioni che il treno visita Ns) ed infine l'overhead generato dal MR per ogni singolo CN del VMN che aggiorna la sua

sessione esistente [Figura 19c]. Questo dipende direttamente dal numero di passeggeri, dal numero unico di CN della rete (Ncn) e dal numero di handover delle stazioni (Nh). Nilnp e Bilnp corrispondono al packet ed al bandwith overhead.

$$C_{ILNP} = \frac{J_1.N_p + J_2.N_s + J_3.N_p.N_{CN}.N_h}{N_p.N_s.T_h}$$

Figura [17 Metro]

$$N_{ILNP} = \frac{8.N_p + 8.N_s + 2.N_p.N_{CN}.N_h}{N_p.N_s.T_h}$$

$$B_{ILNP} = \frac{1362.N_p + 1362.N_s + 144.N_p.N_{CN}.N_h}{N_p.N_s.T_h}$$

Figura [18 Metro]

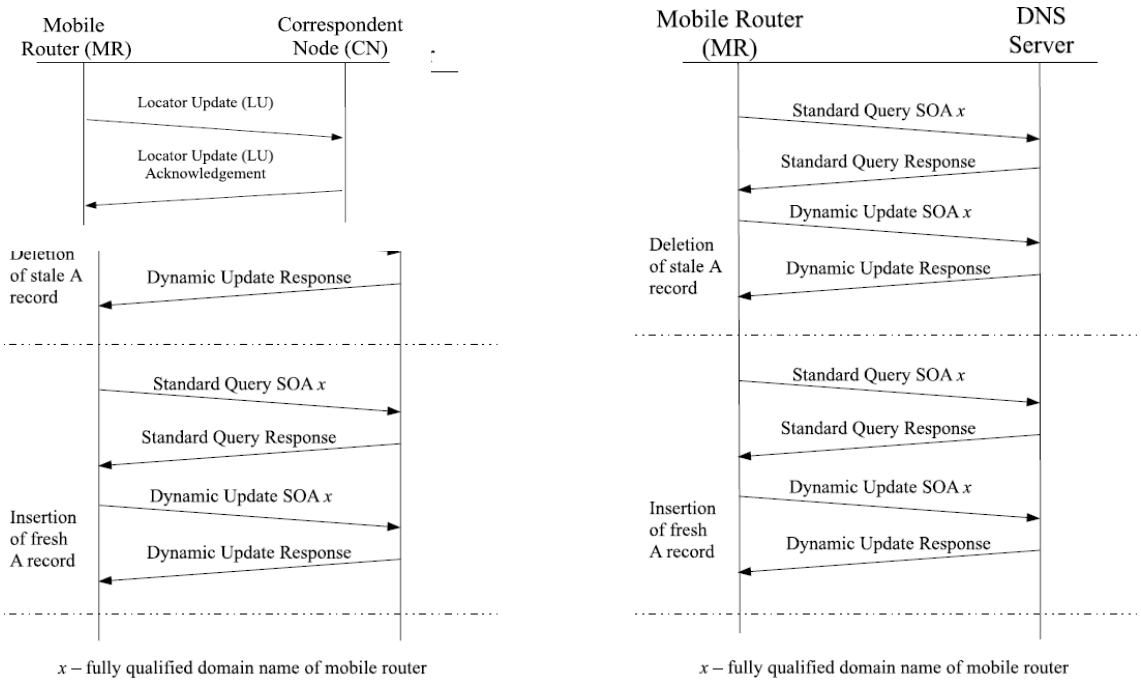


Figura [19 a b c Metro]

a) Inizializzazione VMN: 8 pacchetti, 1362 bytes;

b) MR handover: 8 pacchetti, 1362 bytes;

c) CN update : 2 pacchetti, 144 bytes.

4.4 Risultati

Il protocollo di scambio per ILNPv6 è più semplice di quello di Nemo/Optinets e l'elaborazione che ne risulta anche, visto che Nemo (ad esempio) ha bisogno di due tipi di tunnel. Inoltre, ILNPv6 sfrutta le esistenti infrastrutture DNS per il naming, mentre Nemo/Optinets deve introdurre entità di rete aggiuntive (HA e FA) per funzionare. L'utilizzo dei tunnel inoltre crea potenzialmente un inoltro inefficiente dei pacchetti ed una certa complessità del sistema, se combinato con la redirezione attraverso l'home network. Usando le espressioni di calcolo per l'overhead di pacchetti e di banda, variamo il valore di Nh da 1 a 14 (metà circuito) ed il valore di Ncn da 1 a 20 (Ogni VMN ha 20 CN unici).

Dalla [Figura 20] vediamo che, comparato con Nemo, l'overhead di OptiNets è molto più alto (di un fattore di 10). Dalla [Figura 21] vediamo un simile incremento per OptiNets comparato con Nemo.

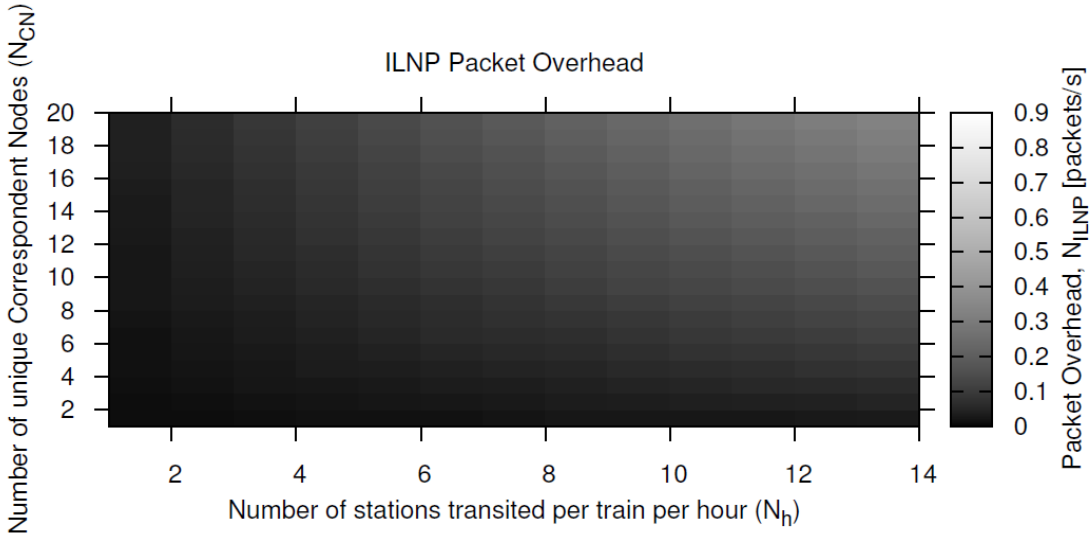


Figura [20 a Metro]

a) ILNPv6 – Equazione Nilnp (min/max: 0.01/0.35)

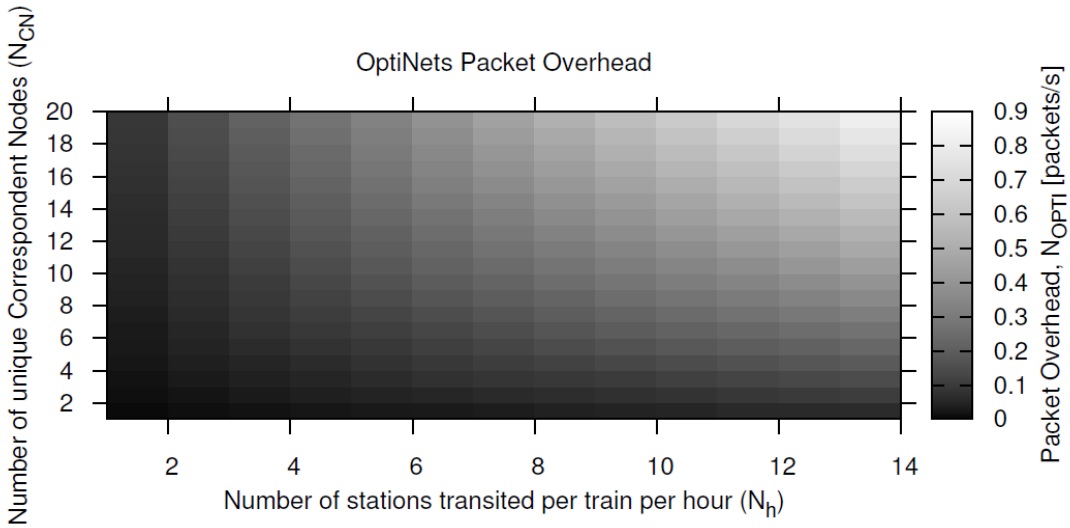


Figura [20 b metro]

b) OPTI - Equazione Nopti (min/max: 0.01/0.87)

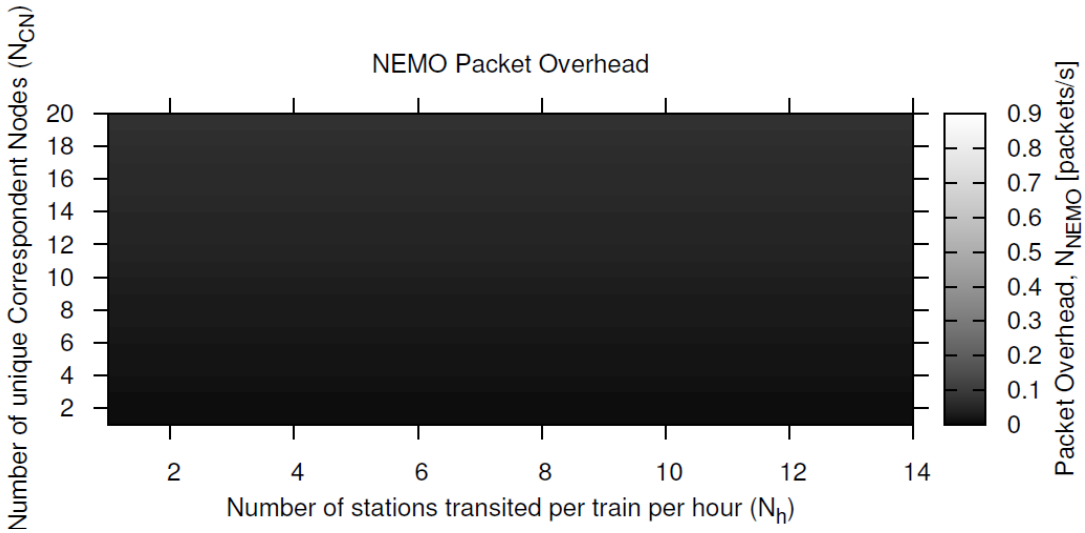


Figura [20 c metro]

c) NEMO - Equazione Nnemo (min/max: 0.01/0.08)

Gli assi orizzontali sono le stazioni attraverso cui un treno è passato per ogni ora, gli assi verticali l'overhead di banda e di pacchetto per persona (bytes). Un'ombra più nera rappresenta minore overhead di banda.

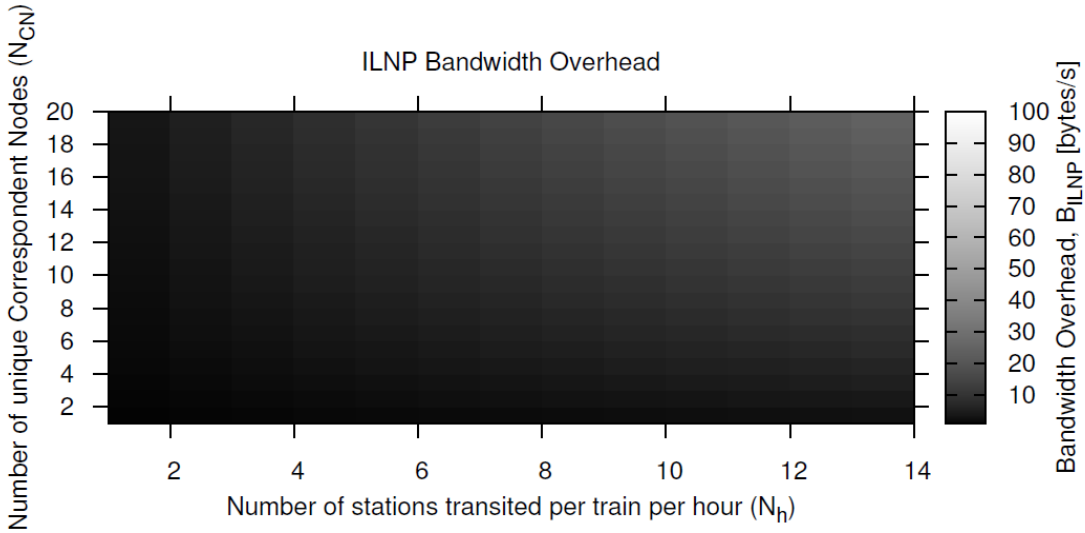


Figura [21 a metro]

a) ILNPv6 - Equazione Nilnp (min/max: 0.99/25.79)

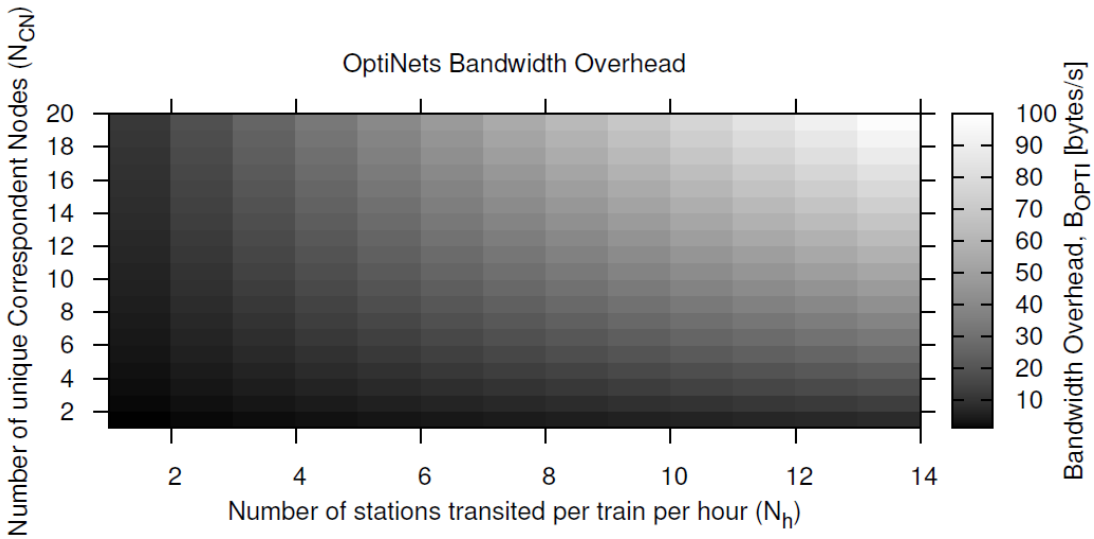


Figura [21 b metro]

b) OPTI – Equazione Nopti (min/max: 0,57/104,59)

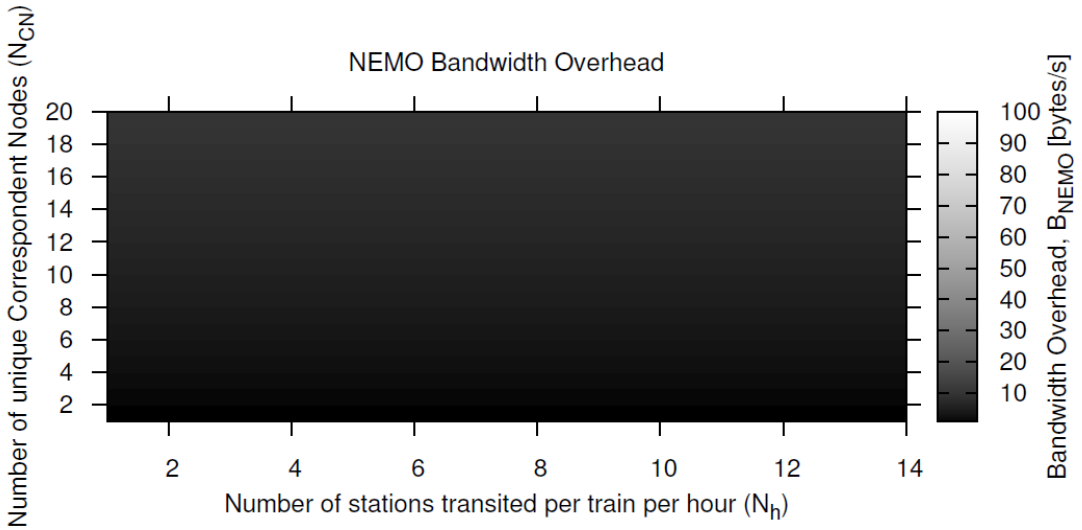


Figura [21 c metro]

c) NEMO – Equazione Nnemo (min/max: 0,70/10,51)

Gli assi orizzontali sono le stazioni attraverso cui un treno è passato per ogni ora, gli assi verticali l'overhead di banda e di pacchetto per persona (bytes). Un'ombra più nera rappresenta minore overhead di banda.

Guardando le equazioni generali della prima emulazione, vediamo che rimane un'incognita la durata di ogni singolo viaggio del passeggero e quante stazioni salta. Questo a causa della mancanza di tracce di ogni singolo individuo. Per ovviare, abbiamo assunto una distribuzione statistica casuale di passeggeri (con due CN per ognuno) per ottenere valori di handover per ogni passeggero. Definiamo anche come rapporto di mobilità di passeggeri (Rp), il numero di passeggeri nel treno che rimane a bordo. Per l'emulazione, abbiamo usato tre diversi valori di Rp, 10%, 50% e 90%. In totale quindi è stata eseguita questa emulazione tre volte, ognuna con un diverso valore di Rp. Ognuna emulava un anno di treni e di movimento passeggeri. Quindi abbiamo usato i risultati corrispondenti per aggiornare le equazioni generali di banda per Nemo, Optinets ed ILPNv6, nelle [Figure 22 a b c] rispettivamente.

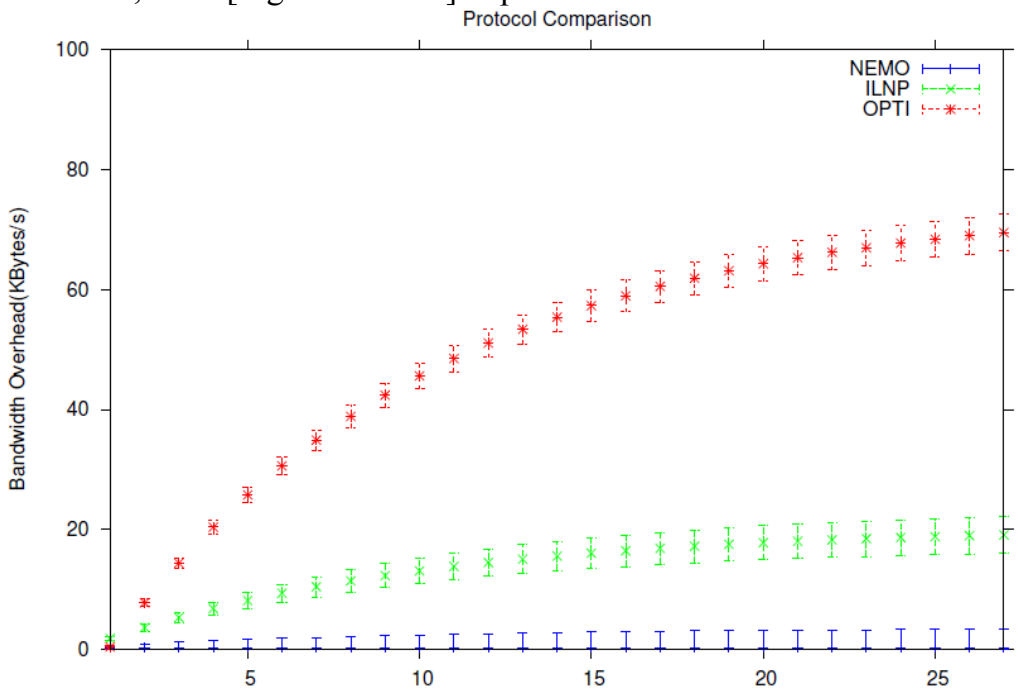


Figura [22 a Metro]

a) 10% Passenger Movement Ratio

Bandwidth overhead per treno [Kb/s] quando CN = 2

Gli assi orizzontali sono le stazioni attraverso cui un treno è passato per ogni ora.

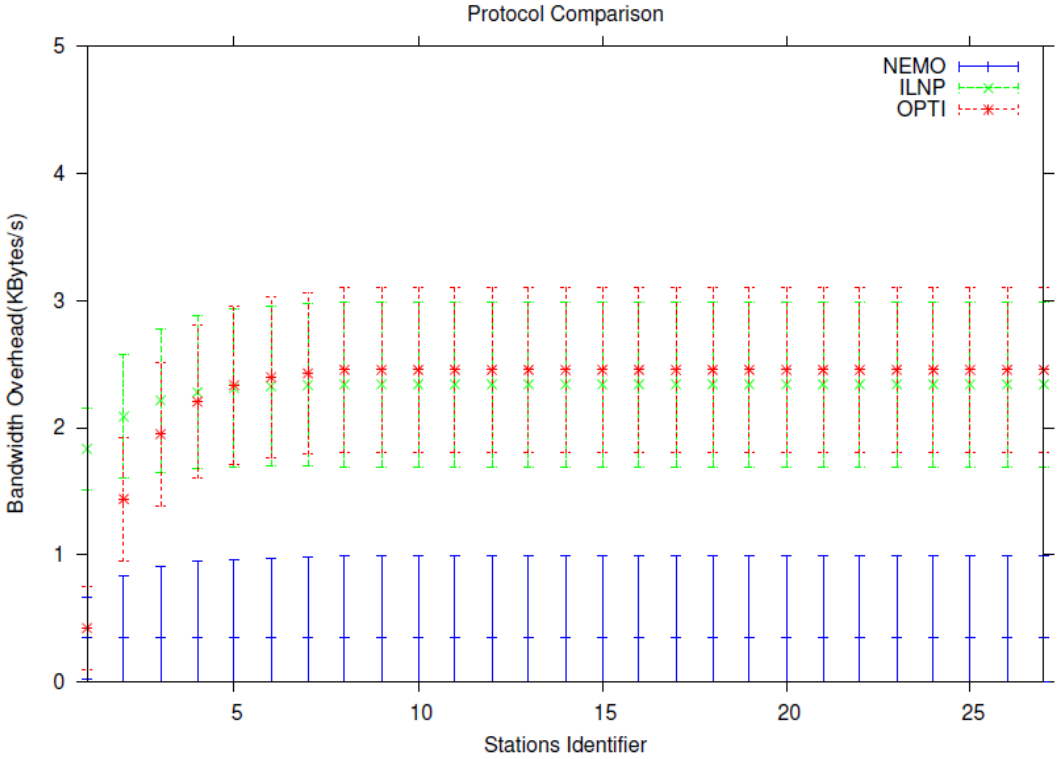


Figura [22 b Metro]

b) 50% Passenger Movement Ratio

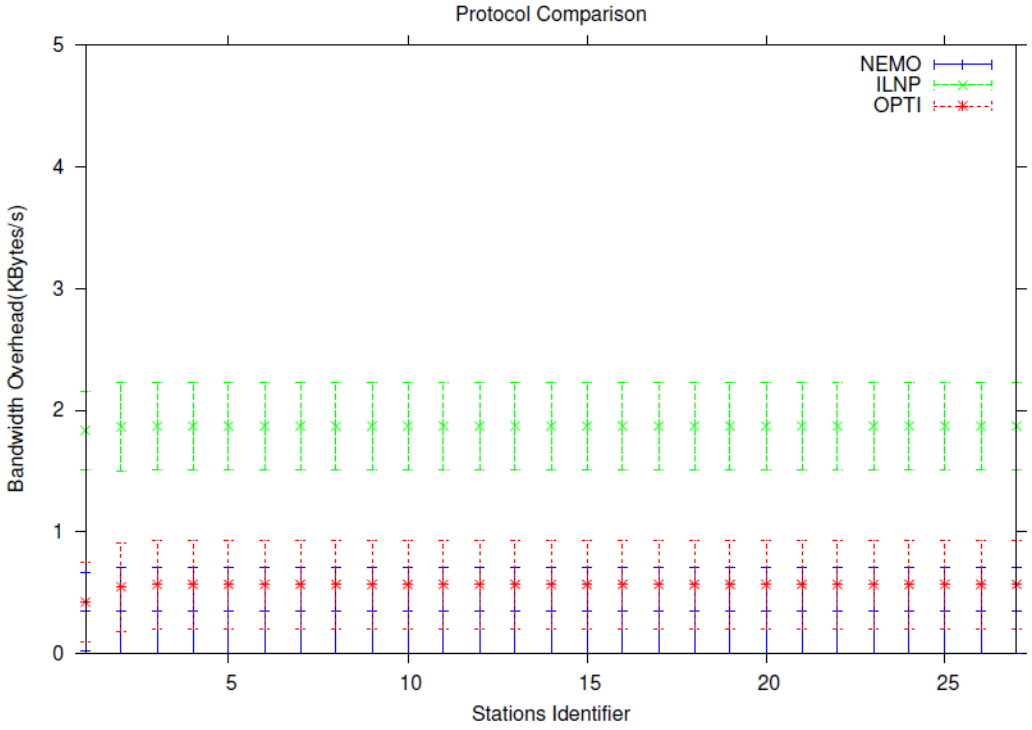


Figura [22 c Metro]

c) 90% Passenger Movement Ratio

Abbiamo notato che questo non ha un effetto visibile per quanto riguarda Nh, poiché dipende direttamente dal tasso di movimento passeggeri Rp, che è coerente con una distribuzione uniforme random. Valori più alti di Rp (più passeggeri che lasciano / entrano nella rete) corrispondono ad un minore numero medio di handover per ogni stazione (valore minore di Nh).
Abbiamo anche notato che, a seconda del tasso di passeggeri che salgono e scendono dal treno, c'è anche uno stato costante dove il numero di handover per stop rimane stabile. Meno differenza c'è tra questi due tassi di ingresso/uscita, più velocemente lo stato costante è raggiunto.
Osservando quanto OptiNets ed ILNPv6 si comportano con diversi valori di Rp, vediamo che ILNPv6 è molto meno sensibile al cambio di numero di handovers (Nh). Quando il Rp è settato a 50% [Figura 22 c], il costo di OptiNets aumenta di un ordine di grandezza rispetto all'incremento di ILNPv6. Come risultato, in uno scenario con grandi fluttuazioni di passeggeri, l'approccio OptiNets probabilmente potrebbe portare a variazioni di utilizzo di banda molto più alte comparato a ILNPv6.

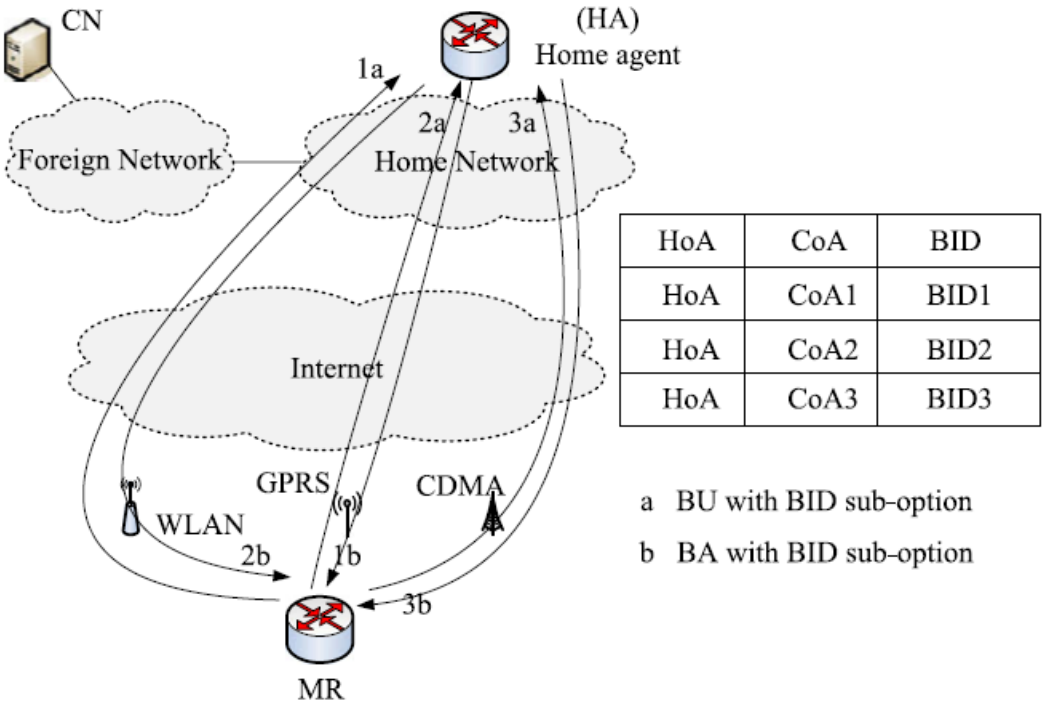
5. SECONDA EMULAZIONE : HANDOVERS

In questo caso vogliamo studiare un MR dotato di tecnologia WLAN, CDMA e GPRS e capace di interfacciarsi contemporaneamente con ognuna di esse. [22]

5.1 PREMESSA

Con Nemo, il Mobile Router serve come gateway; l'indirizzo permanente chiamato Home Address (HoA) è ottenuto come un identificatore dell'MR. Quando l'MR si sposta, acquisisce un care of address (CoA) dall'access router (AR) nel nuovo network (Foreign Network). Il MR manda un binding update (BU) all'home agent (HA) situato nell'home network, legando il CoA con l'HoA. L'HA risponde con un binding Acknowledgemetn (BA). Dopo il binding, il tunnel bidirezionale è stabilito tra il MR e l'HA. I Pacchetti dal corrispondente nodo (CN) con la destinazione HoA dell'MR sono direttamente instradati all'HA, che deve poi reinstradarli al CoA del MR attraverso il tunnel. I nodi network mobili (MNN Mobile Network Nodes) nella rete mobile hanno un indirizzo permanente preso dal mobile network prefix (MNP), pubblicizzato all'ingresso dell'interfaccia MR. I pacchetti da e per il MNN sono incapsulati nel tunnel.
Le WLAN sono le reti wireless più ampiamente utilizzate, possono fornire un bitrate superiore a 11Mbit/s (IEEE802.11b), ma la loro area di copertura del segnale è piccola. Al contrario, le reti CDMA e GPRS possono fornire una grande mobilità ed una connessione attiva costantemente per gli utenti mobili, ma le loro capacità quantitative di trasporto dati sono molto inferiori rispetto alla WLAN.

5.2 IL FUNZIONAMENTO



HoA	CoA	BID
HoA	CoA1	BID1
HoA	CoA2	BID2
HoA	CoA3	BID3

Figura [23 Handover]

Registrazione di CoA multipli

Nel nostro schema, ogni interfaccia attiva del MR configura un CoA e manda un messaggio BU all'HA. Ogni messaggio BU contiene un BID (Binding Unique Identifier) per ogni CoA e ogni CoA è legato al HoA del MR. HA risponde con un messaggio BA contenente differenti opzioni secondarie BID. Quindi il triplo tunnel bidirezionale è stabilito tra MR ed HA.

Solitamente, quando una WLAN non è disponibile, il MR si connette ad Internet tramite la rete CDMA/GPRS (ovviamente in presenza di segnale). Per abilitare l'handover senza perdita di dati tra le interfacce, amplio la l'opzione secondaria BID [Figura 24] inclusa nei messaggi BU e BA.

In [Figura 24], BID è stata già definita per identificare i diversi bindings. Aggiungiamo le flag “D” e “pre_default_BID” come definito in [23]. D è usato per indicare l'evento handover, mentre “pre_default_BID” è usata per

memorizzare il BID dell'interfaccia usata precedentemente. La decisione dell'handover di interfaccia del MR è mostrato in [Figura 25] ed è spiegato dettagliatamente nei prossimi paragrafi.

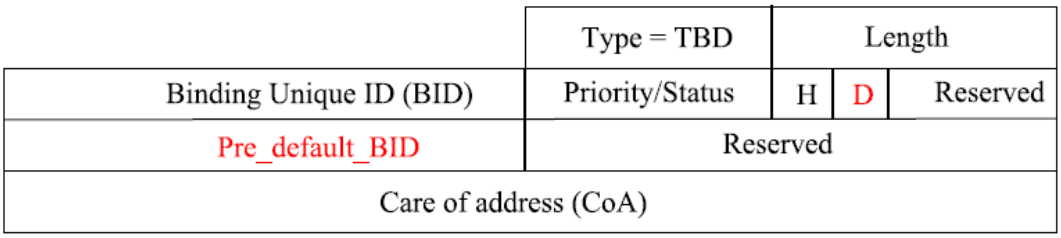


Figura [24 Handover]

Modifica dell'opzione secondaria

Come mostrato da [Figura 25], “Delay for t seconds” indica che le interfacce WLAN eseguono scansioni alla ricerca del segnale dell'access point (AP) corrente ad ogni intervallo di t secondi. Se la forza del segnale è minore di Sth, si cercherà un altro AP migliore. Se non esiste nessun AP, MR cambierà interfaccia in CDMA/GPRS. T secondi più tardi, verrà rieffettuata la scansione del segnale. Se il traffico cambia da un'interfaccia all'altra, un trigger manda un messaggio alla memoria condivisa, informando “l'indice delle interfacce” di quella nuova in uso, che sarà poi letta del protocollo MCoA NEMO. Ricevendo questa informazione, il MR cambia l'opzione secondaria BID ed il campo priorità e notifica all'HA l'handover sempre attraverso il protocollo McoA.

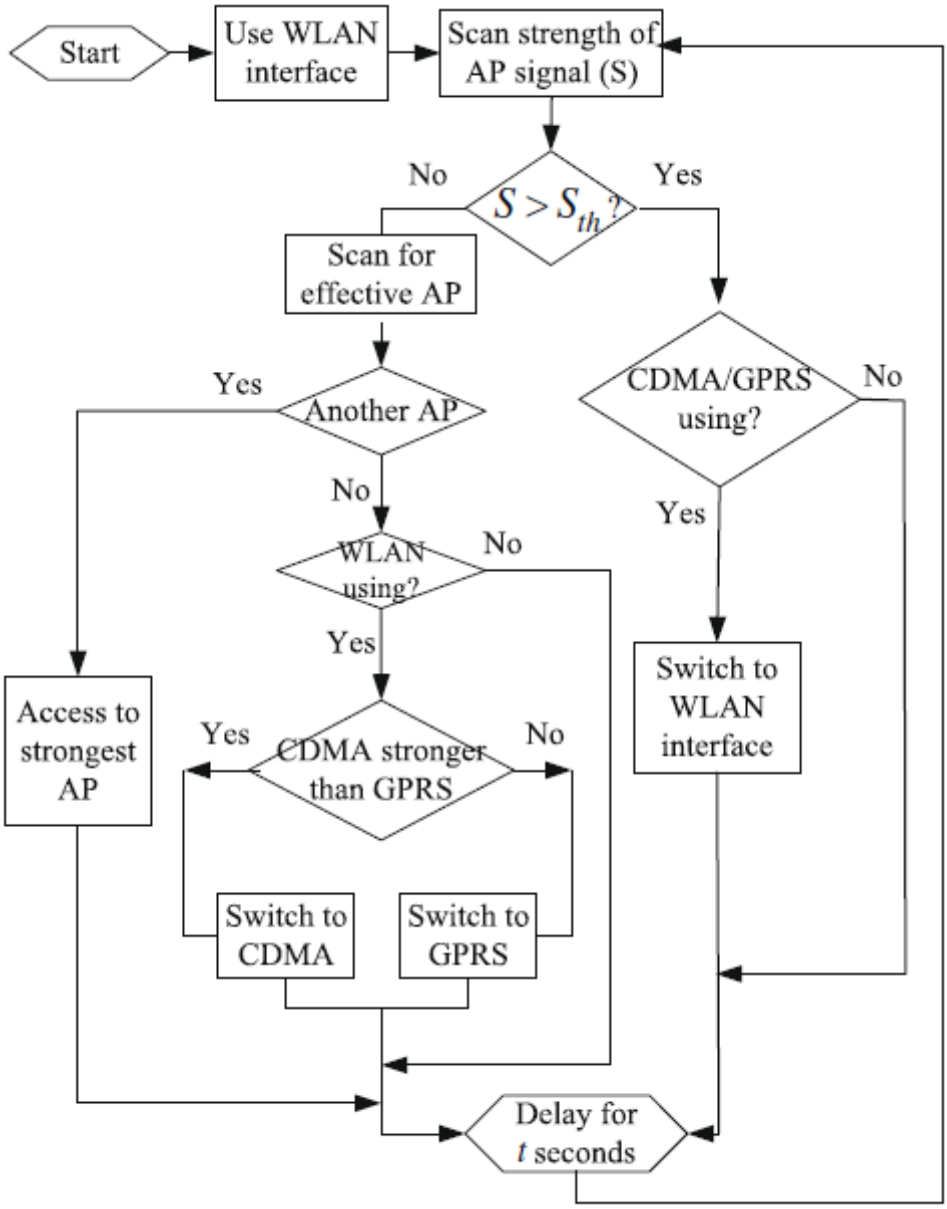


Figura [25 handover]

Meccanismo decisionale per gli handovers.

C'è un solo BID per ogni interfaccia, quindi il MR può mandare la notifica all'HA semplicemente cambiando il BID stesso. Siccome il messaggio BU è aggiornato ad intervalli fissi, per evitare il ripetersi di frequenti operazioni da parte di HA, il bit “D” nelle opzioni secondarie di BID è usato per indicare il verificarsi dell'handover. Il campo “pre_default_BID” è usato per immagazzinare il BID in

BU mandato prima. Se “BID” usa un nuovo BU e quello usato prima sono gli stessi, D è settato a 0, che significa che nessun handover è accaduto (viceversa viene settato ad 1).
Ricevendo il messaggio BU dal MR, HA controllerà il bit “D”. Se D = 0, non accadrà nulla. Altrimenti, HA confronterà i campi “pre_default_BID” e BID”, settando la priorità dell'interfaccia corrispondente al BID più alto tra le due interfacce. Quindi tutti i pacchetti sono inviati alla nuova interfaccia.
Tutto questo processo è trasparente allo strato applicativo, quindi è possibile misurare le varie performance durane gli handover per valutarle.

5.3 I RISULTATI

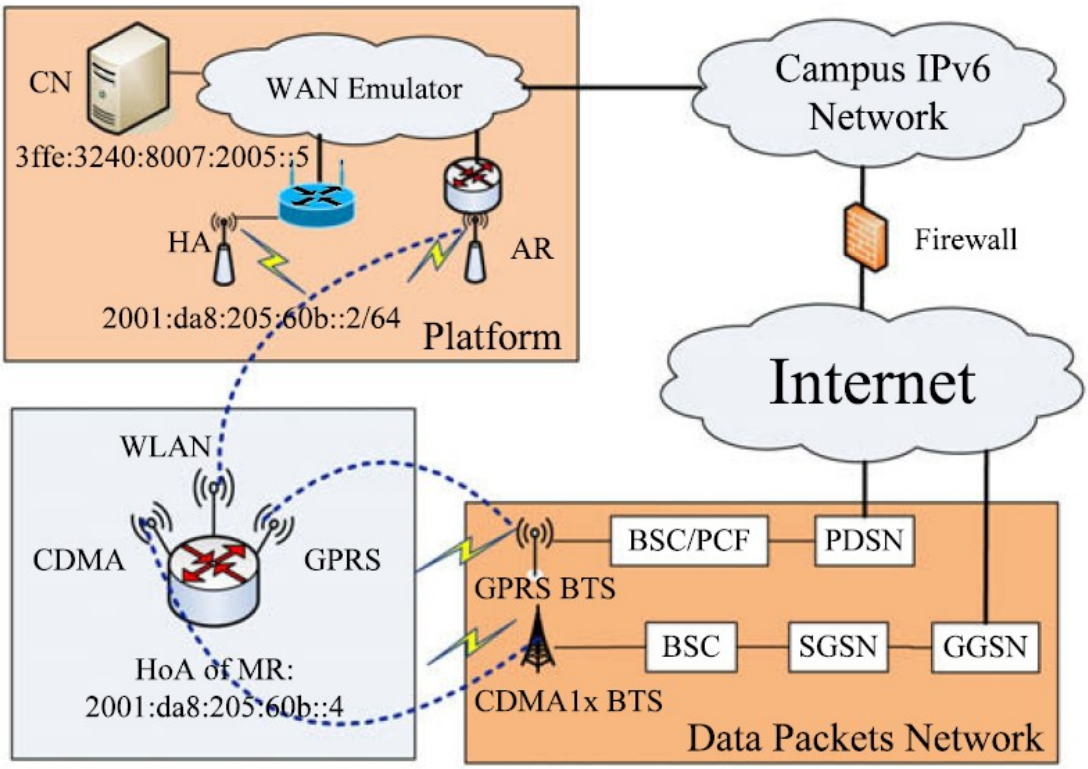


Figura [26 handover]

La configurazione di rete usata nell'esperimento è quella nella [Figura 26].
Cn e HA per il MR sono situati all'home network IPv6 del Campus. La WLAN è stata settata con l'IPv6 nativo, così può configurare un indirizzo globale da solo. Così, un tunnel IPv6-IPv4 è configurato tra il MR e l'AR nella rete CDMA ed un IPv6 in UDP tra il MR e l'AR in GPRS.
La prestazioni della consegna dei pacchetti sono valutate misurando il tempo di RTT dei diversi collegamenti costituiti tra il MR ed il CN con le interfacce WLAN, CDMA e GPRS. I valori di RTT tra il MR ed il CN mandando 56 bytes ICMPv6 e risposta con "ping6" attraverso le varie tecnologie sono schematizzati nella [Tabella 3], da cui possiamo notare che l'interfaccia che ha prestazioni migliori è la WLAN. Al contrario la GPRS è la peggiore a causa del suo data rate.

	Packet loss ratio (%)	RTT_min (ms)	RTT_avg (ms)	RTT_max (ms)
WLAN	5	38.152	50.352	1516.344
CDMA	10	159.477	618.037	4140.511
GPRS	57	878.883	4131.835	5656.169

[Tabella 3]

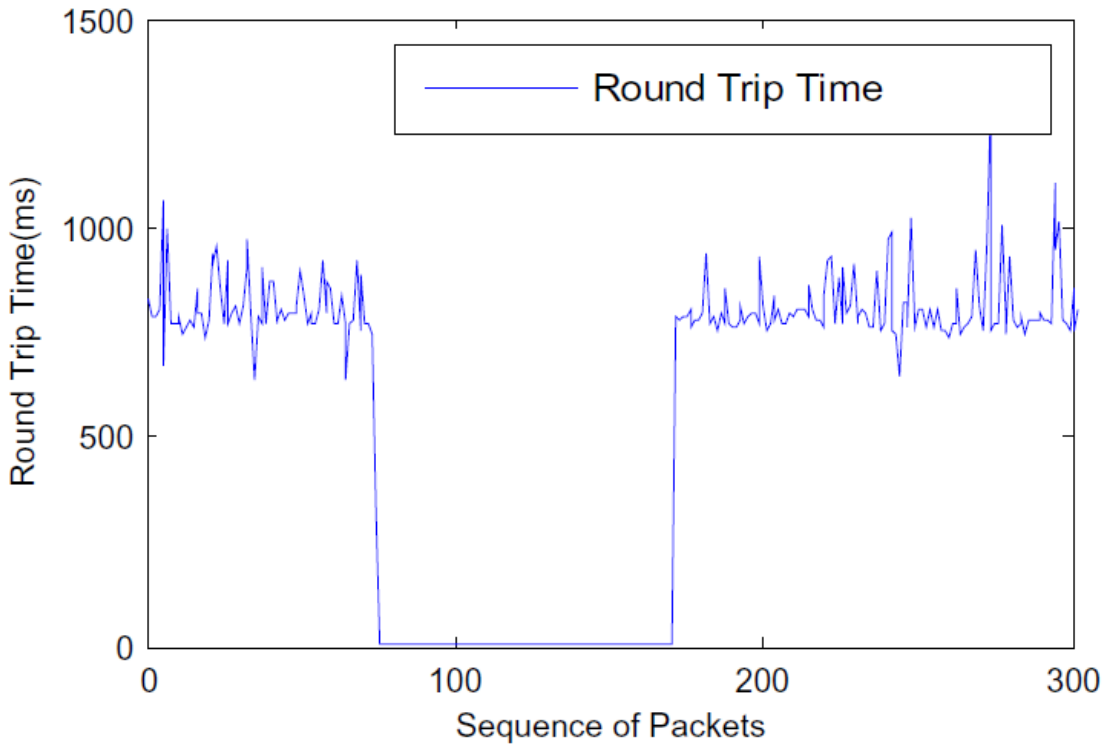


Figura [27 handover]

La [Figura 27] mostra la durata media dell'handover tra WLAN e CDMA. Il RTT è misurato ogni secondo mandando 56 bytes ICMPv6 e rispondendo con "ping6". Il tempo medio è di 3 ms (poiché la WLAN è in scala piccola nell'esperimento), mentre quella del CDMA sui 600 ms. Possiamo vedere che il traffico trasferito da WLAN a CDMA è scorrevole, senza pacchetti persi a causa del data rate di entrambi, sufficientemente buono per il nostro test.

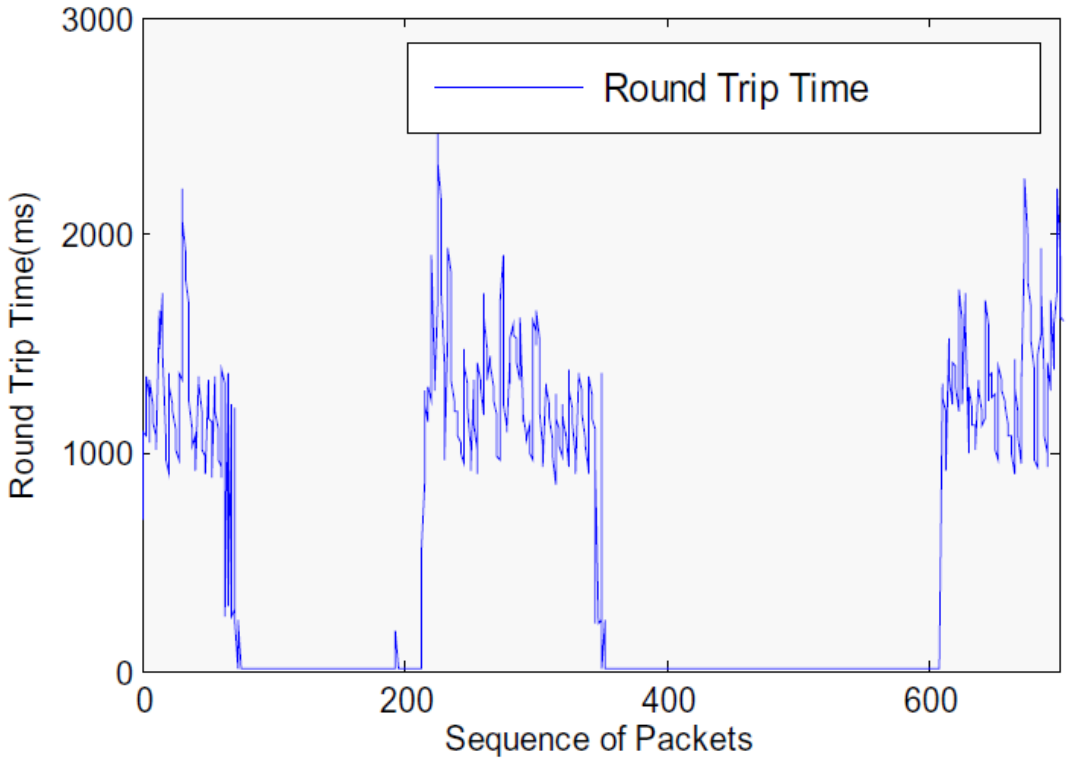


Figura [28 handover]

Una simile operazione è stata fatta successivamente, mandando gli stessi pacchetti tra WLAN e GPRS. Come si può vedere nella [Figura 28], nessun pacchetto è stato perso durante l'handover, a causa della rapidità con cui è stato svolto. Il RTT del GPRS è di 1000-2000 ms. I pacchetti persi durante l'intera trasmissione sono stati circa il 15%, causato dalla relativamente bassa qualità del collegamento GPRS nel nostro ambiente di sperimentazione. Quando è stato usato CDMA, nessun pacchetto è stato perso.

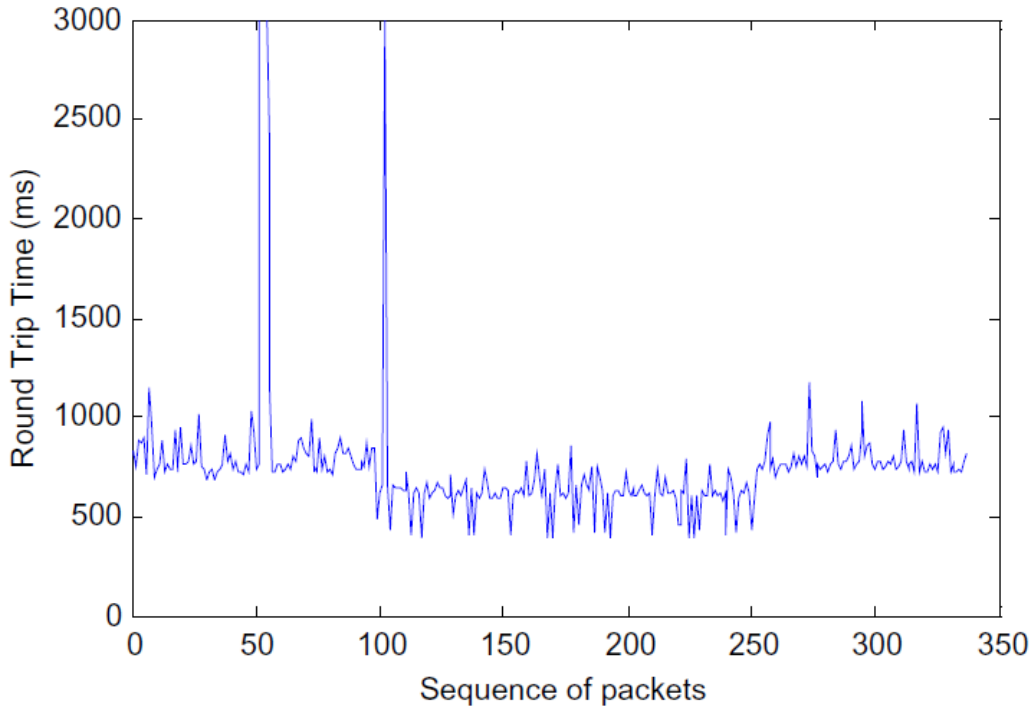


Figura [29 handover]

La [Figura 29] mostra le performance tra GPRS e CDMA, con gli stessi messaggi mandati nei casi precedenti. Il traffico trasferito da GPRS a CDMA è senza pacchetti persi, a causa del loro relativamente basso tempo di handover. Possiamo quindi vedere dagli esperimenti appena svolti che possono essere usati tre tunnel bidirezionali ed indipendenti contemporaneamente. Il MR li gestisce indipendentemente dal loro traffico.

5.3 ANALISI PERFORMANCE

Le due problematiche fondamentali durante gli handovers sono il tempo di interruzione del servizio e la quantità di pacchetti persi. Per l'analisi delle performance, noi utilizziamo i seguenti parametri:
_latenza totale handover (th), composta da quanto tempo viene impiegato per

identificare il movimento (t_{md}), il ritardo di configurazione del CoA (t_{coa}) ed il ritardo di registrazione (t_{re}).
Il processo di handover standard e la sua latenza sono mostrati nella [Figura 30]

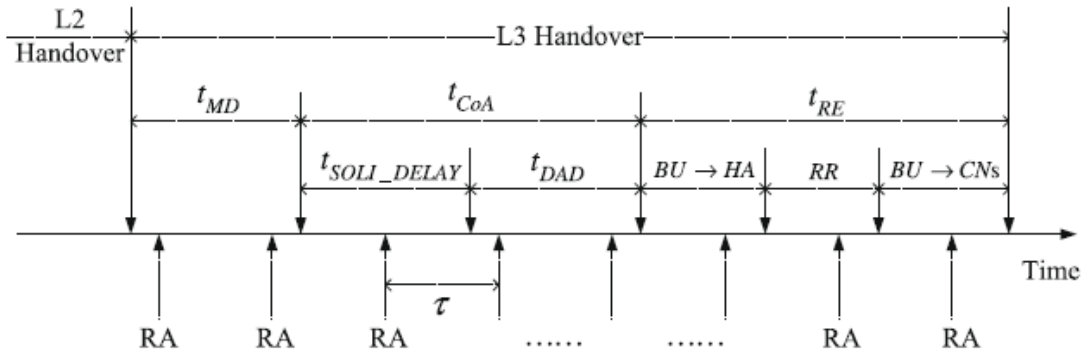


Figura [30 handover]

Il MR si accorge del movimento ricevendo un messaggio di tipo router advertisement (RA). L'intervallo RA è un valore random compreso tra MinRtrInterval e MaxRtrInterval che nei casi reali, per evitare troppo overhead di segnali, è di qualche secondo. In questo caso sono settati a 30 e 70ms.
Dopo aver ricevuto RA, il MR configura CoA, secondo il prefisso che ha acquisito da quel messaggio. Per evitare il duplicarsi degli indirizzi, viene fatto aspettare appositamente un certo tempo e avviato un “duplicate address selection” (DAD). Le specifiche MIPv6 suggeriscono che questo tempo sia tra 0 e MAX_RTR_SOLICITATION_DELAY, settato a 1000ms. Quindi $E(t_{soli\_delay})$ è 500ms. Un'altra parte del t_{coa} è il ritardo causato dall'esecuzione del DAD. Si può calcolare in 1000 ms.
'ultima parte dell'handover del MR è il processo di registrazione. Prima di tutto il MR deve notificare al suo HA il nuovo CoA. La più grande latenza di questo processo è causata dal ritardo di arrivo dei messaggi (correlato alla distanza tra il MR ed il HA). In totale, la latenza dell'handover è calcolata come

$th = E(t_{md}) + E(t_{soli\_delay}) + RETRANS\_TIMER + t_{re.}$

Secondo i calcoli fatti precedentemente, il ritardo dell'handover standard di NEMO è superiore a 1,5s e quindi non è utilizzabile da servizi particolarmente sensibili a questo dato.
La quantità di pacchetti persi durante gli handover può essere espressa in percentuale sul rapporto totale di pacchetti mandati.
Le [Figura 31] e [Figura 32] comparano il tempo di interruzione di servizio e la quantità di pacchetti persi tra il MR multi interfaccia proposto ed un MR a singola interfaccia in NEMO BSP. Assumiamo che l'intervallo di router advertisement sia di 4 s; il raggio di una cella AR 1 km; quando un MR è lontano dall'HA dell'home network e connesso ad una rete straniera via AR, il RTT tra il MR e l'AR sarà molto più piccolo rispetto a quello tra l'AR e l'HA. Questo perchè quando un network mobile lascia il suo Home Network, il MR si attacca all'AR direttamente, quind il RTT tra MR e AR è piuttosto piccolo nonostante c'è una distanza relativamente lunga tra il MR e l'HA del suo home network.
Nella [Figura 32] settiamo la velocità di movimento del router con valori tra 50 e 300 km/h, in modo da simulare la velocità di un veicolo. La quantità di pacchetti persi varia tra l' 1 ed il 9%, in maniera incrementale con l'aumento di velocità. Nessun pacchetto è perso, visto che CDMA e GPRS hanno una copertura ampia e possono essere usati contemporaneamente.

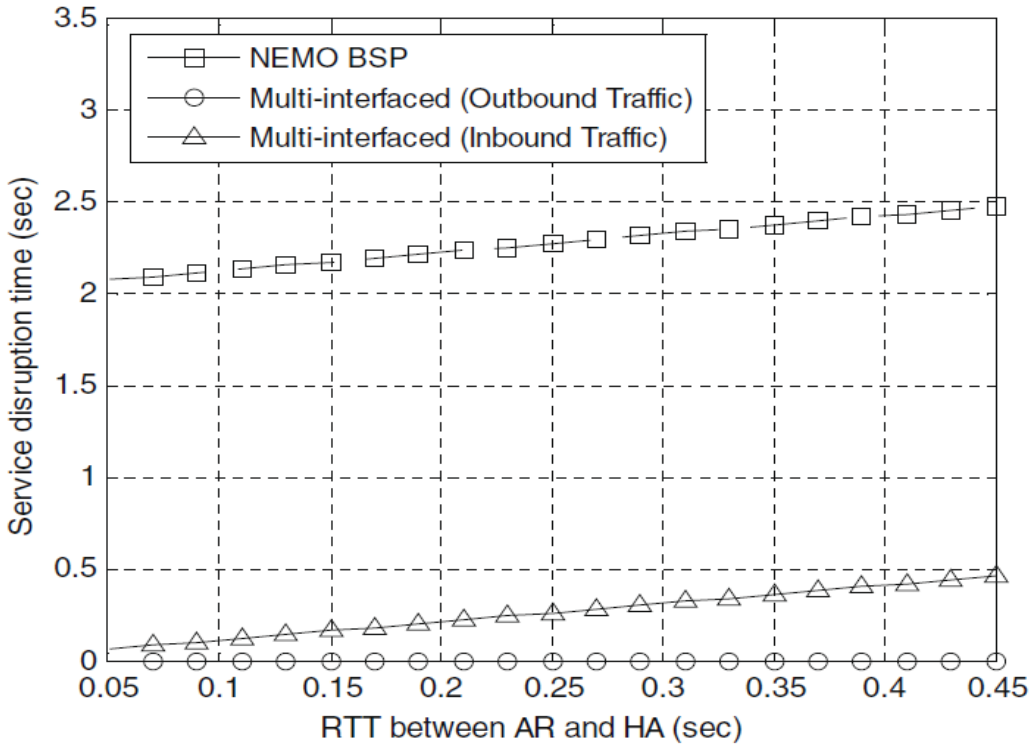
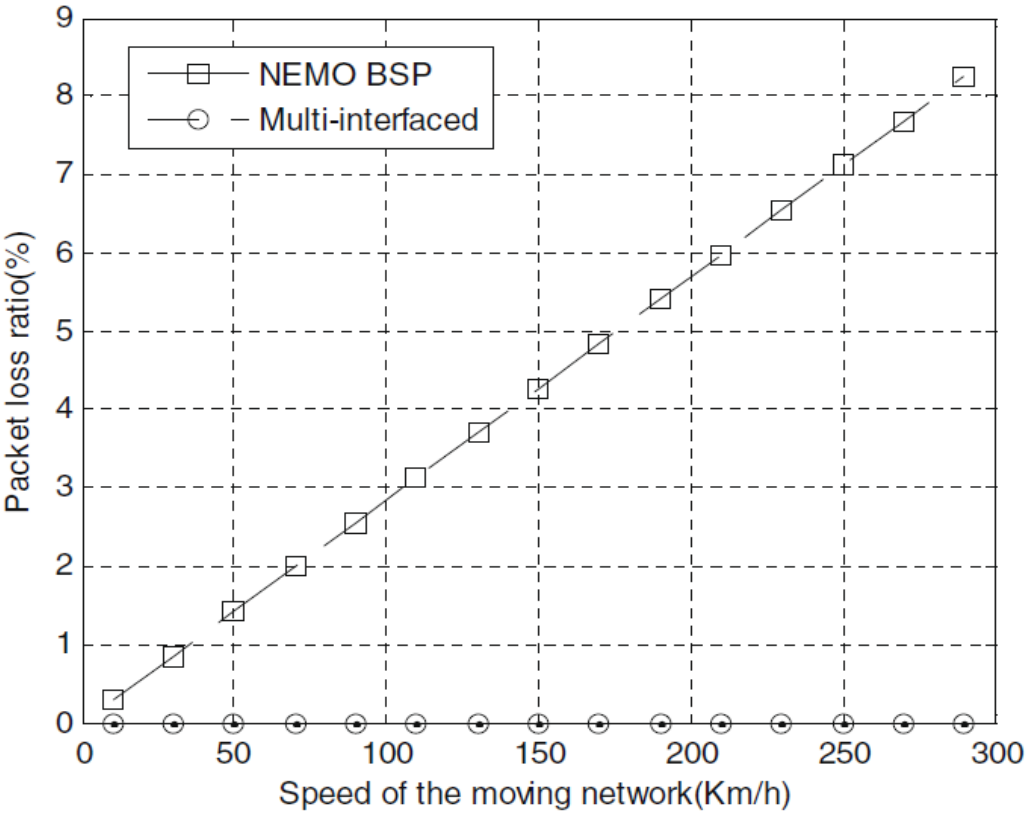


Figura [32 handover] : Interruzione di servizio vs diversi RTT AR - HA

Figura [33 handover] Quantità di pacchetti persi vs movimento rete



E' ovvio che MIPv6 e NEMO BSP non sono la giusta scelta se si ha bisogno di utilizzare una rete che sia particolarmente sensibile alla latenza (superiore a 1,5s). I risultati indicano che il traffico può essere trasferito senza perdite da un'interfaccia all'altra, ma anche con un'interfaccia sola cambiando posizione in una rete eterogenea. Questo schema può essere applicato ai nodi mobili, specialmente quando si muovono in aree ampie e possono usare diversi ISP.

6. CONCLUSIONI

6.1 Future Wireless Communications

L'ambiente del futuro, per quanto riguarda la connettività wireless, è caratterizzato dalla coesistenza di molte diverse tecnologie network, come LTE, LTE-Advanced, UMTS, WiMAX, WiFi ecc, altamente sovrapposte e sovrapponibili, con caratteristiche tecniche e di trasmissione molto diverse.
Gli operatori di rete dovranno tenere conto della diversificazione delle richieste, della loro tipologia e della scelta della risorsa di collegamento specifica più efficiente per completare il collegamento.
Questo meccanismo, chiamato “access network selection mechanism” prevede che una rete è scelta per far comunicare il dispositivo mobile con internet o con altri dispositivi mobili ed un handover ad una tecnologia migliore è eseguito dinamicamente in base alla QoS ed alle richieste dell'utente.
In maniera complementare a questo, serve un meccanismo di scoperta dei punti di accesso alla connessione di prossimità.
Attualmente il meccanismo di selezione diffuso prevede che sia l'utente che decide indipendentemente l'accesso attraverso il quale comunicherà, ma questo impedisce una visione più globale della situazione, poiché il nodo mobile non è a conoscenza del contesto momentaneo delle reti, quindi questo meccanismo non offre garanzie che la scelta effettuata sia in grado di sostenere la comunicazione in maniera appropriata e continuativa.
Anche per il meccanismo di scoperta delle reti, le soluzioni correnti sono per lo più di eseguire scansioni dell'ambiente wireless, un'operazione che consuma molta energia.
Con l'evoluzione delle reti auto-organizzate e le Cognitive radio Technology,

l'access network può essere in grado di adattare la potenza di trasmissione alle caratteristiche dell'ambiente circostante, facendo diventare la raccolta delle informazioni ancora più complicata.
Per risolvere questi problemi, NGMN Alliance [9] indica un meccanismo che cerca di bilanciare il servizio tra costi ed efficienza e le performance che sono richieste dall'utente e le risorse disponibili tra i diversi access network. [24]

6.2 Altre conclusioni personali

La quantità di documentazione trovata relativa a questo specifico ambito delle reti è stata ampissima. L'immediato vantaggio commerciale, industriale ed il prestigio accademico che può portare l'implementazione di un sistema continuo e veloce di scambio dati attraverso diverse interfacce e tecnologie di comunicazione è facilmente comprensibile ed appare nella notevolissima mole di proposte che, anche soltanto per piccoli problemi relativi, è stata pubblicata.
Le soluzioni proposte per permettere un cambio di gestore di rete si scontrano tutte con la questione della sicurezza.
Le questioni di scelte infrastrutturali, lato “provider” sono rilevantissime. A seconda del posto e dell'utilizzo che si vuole fare di questo servizio, la tipologia di antenne ed il software di utilizzo possono essere diversi sotto moltissimi punti di vista, tenendo conto di tecnologie, non trattate qui ma in via di perfezionamento, che tentano di prevedere quale sarà la prossima “cella” o “rete” dove un nodo mobile potrebbe finire in base a questioni di prossimità.
Come scritto precedentemente spiegando il funzionamento di Nemo, la strada per ora migliore e più percorsa per far riconoscere ad un diverso ISP un cliente o un utente è quella di appoggiarsi a server di fiducia esterni, rallentando e creando overhead nella connessione. Questo porta anche ad un aggravio di installazione

di nuovo hardware e software nella strumentazione esistente, ma è il tradeoff più semplice da valutare, visto che una delle alternative sarebbe il dover modificare i protocolli comunemente più usati nel web.
L'utilizzo di gateway proprietari proposto da Cisco con LISP è il caso più evidente e forse uno dei più famosi. Può prendere facilmente piede per una questione di facilità di schema e praticità nel caso gli ITR ed ETR vengano iniziati ad essere usati in maniera pesante e comune nelle reti di tutti i tipi.
Il maggior problema introdotto da uno schema che separa Loc ed ID come LISP è che quando un nodo si sposta, c'è bisogno di cambiare tutto il mappaggio tra l'EID ed un certo numero di RLOC nella sua nuova posizione nella rete. Quando questo è aggiunto all'overhead prodotto dai bindings update di MIPv6, qualche pacchetto può essere ritardato o perso. In generale, una questione che riguarda tutte le tecnologie viste, è trovare un buon bilanciamento con il tasso di aggiornamento della posizione del nodo da controllare, poiché in assenza di questo vengono continuati ad essere spediti pacchetti praticamente persi.
Le questioni principali con cui discernere un buon sistema sono tutte relative alle interruzioni di servizio durante gli handover ed all'overhead causato da quest'ultimo. Eseguire o rapportarsi coi molti test eseguiti e facilmente trovabili tra la documentazione scientifica (dipendenti dal numero di nodi in movimento, rapportati con il traffico che si prevede o il lavoro che si intende fare) costituiscono la miglior strada per avere un servizio efficiente aspettando una soluzione più completa ed uniforme.

7. BIBLIOGRAFIA

[1] http://en.wikipedia.org/wiki/Mobile_phone

[2] “Energy-Efficient Mobile Data Uploading from High-Speed Trains” Xiaoqiang Ma · Jiangchuan Liu · Hongbo Jiang

[3] “A Unifying Perspective on Context-Aware Evaluation and Management of Heterogeneous Wireless Connectivity” Bellavista, P.; Corradi, A.; Giannelli, C.; 2010

[4]http://searchnetworking.techtarget.com/definition/home-agent

[5]https://tools.ietf.org/html/rfc4866

[6]http://en.wikipedia.org/wiki/Multihoming

[7]http://en.wikipedia.org/wiki/IP_in_IP

[8] “Elimination of Generalized Ping-Pong Effects Using Triple-Layers of Location Areas in Cellular Networks”
Guangbin Fan1, Ivan Stojmenovic2 and Jingyuan Zhang3

[9]http://www.ngmn.org;

[10] http://tools.ietf.org/html/draft-whittle-sram-ip-forwarding-01#section-4.2

[11] “A Secure Deployment Framework of Nemo (Network Mobility) with Firewall Traversal and AAA Server”
Seong Yee Phang, HoonJae Lee , Hyotaek Lim

[12] "A Mobile IPv6 firewall traversal scheme integrating with AAA",
Pan Jian Li, Chen Shan Zhi,Wireless Communications, Networking and Mobile Computing 2006 - WiCOM International Conference, September 2006.

[13] http://www.cisco.com/web/about/ac123/ac147/archived_issues/ipj_11-1/111_lisp.html

[14]http://dl.acm.org/citation.cfm?id=RFC3588

[15]”A study of multimedia application performance over Multiple Care-of Addresses in Mobile IPv6“ Sousa, B. ; Pentikousis, K. ; Curado, M. ; CISUC,

Univ. of Coimbra, Coimbra, Portugal
[16] “Improving the Resilience in IP networks,” G. Schollmeier, J. Charzinski, A. Kirstadter, C. Reichert, K. Schrodi, Y. Glickman, and C. Winklerin High Performance Switching and Routing, 2003, HPSR. Workshop on, June 2003, pp. 91–96.
[17] “Deploying Reliable IPv6 Temporary Networks thanks to Nemo Basic Support and Multiple Care-of Addresses Registration,” R. Kuntz, 2007 International Symposium
[18] “A Comparative Assessment of Routing for Mobile Networks” Devan Rehunathan, Saleem Bhatti – University of St. Andrews
[19] “Subnetwork Encapsulation and Adaptation Layer” Templin, F., , draft-templin-seal-02.txt . Work in progress.
[20] Tubeprune, London underground statistics. Www.trainweb.org/tubeprune/Statistics.htm
[21] T for London. Circle line facts. Www.tfl.gov.uk/tfl/corporate/modesoftransport/tube/linefacts/?line=circle.
[22] “Experimentation and performance analysis of multi-interfaced mobile router scheme” Xiaouhua Chen, Hongke Zhang, Yao-Chung Chang, Han-Chieh Chao september 2009
[23]”Draft-ietf-monami6-multiplecoa-14, Multiple Care-of Addresses Registration” R. Wakikawa, T. Ernst, K. Nagami, V. Devarapalli, , May 28, 2009.
[24] “Access Network Discovery and Selection in the Future Wireless Communication” Corici, Fiedler, Magedanz, Vingarzan 2011

8. RINGRAZIAMENTI

Il primo ringraziamento è sicuramente dedicato al Prof. Vittorio Ghini, che con la sua vastissima conoscenza dell'argomento, la competenza e la grandiosa umanità ha contribuito in maniera decisiva a darmi la forza di completare questo percorso e scrivere la tesi.
E' conseguenza diretta ringraziare tutta la mia famiglia, che ha sostenuto questi anni di studio materialmente, moralmente e senza mai mollare.
Sarei un ingrato se dimenticassi subito dopo di loro il contributo decisivo, importante e fondamentale di Matteo, Alessandro e Fabio, colleghi di studio, di progetto e di lavoro che a suon di calci, caffeina e pastoni hanno contribuito per la gran parte a conseguire questo risultato.
Quindi vorrei nominare una ad una tutte le persone che mi sono state vicine e che ho conosciuto durante questi anni, ma non ci sarebbe spazio. Sentitevi tutti tirati in causa quindi.
Particolarmente mi sento in dovere di nominare tutti i coinquilini di questi anni, i compagni di idee e di fede di qualsiasi tipo, le persone con cui ho viaggiato e con cui ho studiato, le persone che ho sentito parlare e quelle che han cantato per me; quelli dell'Abbaino e questi dell'Ercolani. Koletz, Alex, Blasto, Matty, Panzer, Luke, Miika, Cri, David, Mihi, Zuzzy, Yulai, Marit Elisabeth, Venke, Bryndis, Kristina, Paola, Mauro, Judith, Heine, Anders, Øystein, Martin, Joonas, Hannes, Ilkka, Minttu, Anne C., Saga, Roddy, Karol, Ambra, Chiara, Dado, Laura, Grace, Mouu, Lok, Marco T., Corrado, Tommy, Daniela C&A, Mr. Pedersen, il Fu, LMP and last but not least Francesca, Enrico, Antonello e Michele.