UNIVERSITÀ DEGLI STUDI DI ROMA "TOR VERGATA"

FACOLTA' DI INGEGNERIA DELLE TELECOMUNICAZIONI

DOTTORATO DI RICERCA IN INGEGNERIA DELLE TELECOMUNICAZIONI E MICROELETTRONICA

CICLO XXI

MOBILITY MANAGEMENT IN NEXT GENERATION NETWORKS

Andrea Polidoro

A.A. 2008/2009

Docente Guida/Tutor: Prof. Stefano Salsano

Coordinatore: Prof. Giuseppe Bianchi

Ad Alessandra, a nostro figlio (ancora nella sua pancia) e alla nostra futura famiglia

Abstract (English)

The ITU-T definition of Next Generation Networks includes the ability to make use of multiple broadband transport technologies and to support generalized mobility. Next Generation Networks will need to integrate several IP-based access technologies in a seamless way. In this Thesis, we first describe the requirements of a Mobility Management scheme for multimedia real-time communication services; then, we report a survey of the Mobility Management schemes proposed in the recent literature to perform vertical handovers between heterogeneous networks. Based on this analysis, we propose an application-layer solution for Mobility Management called MMUSE (Mobility Management Using SIP Extension), which is based on the SIP protocol, and satisfies the most important requirements demanded for a proper implementation of vertical handovers. We also implemented our proposed solution, testing it on the field and proving its overall feasibility and its interoperability with different terminals and SIP servers.

In our work we also report measurements results which analyze the performance of the solution in a real world environment, using commercial access networks (WiFi, 3G)

Finally we discuss a methodology for performance evaluation of the solutions for vertical handovers previously described. The performance evaluation is based on simple analytical models and covers both the ideal case (no packet loss) and the real case where there is a given packet loss rate. The methodology is applied to a comparison among three solutions, namely MIPv4, classical SIP mobility management using re-INVITE messages and the proposed MMUSE SIP based solution.

Abstract (Italiano)

La definizione che l'ITU-T fornisce delle "Next Generation Network" (NGN) include la possibilità di poter utilizzare differenti tecnologie di accessi a banda larga e di supportare la mobilità. Inoltre le reti NGN hanno la necessità di integrare le diverse tecnologie di accesso alla rete IP in maniera del tutto trasparente. In questa tesi dapprima si definiscono i requisiti che una soluzione per la gestione della mobilità nei sistemi di comunicazione real-time dovrebbe avere. Successivamente si analizzano le soluzioni per la gestione della mobilità

presenti in letteratura. A partire da quest'analisi nasce MMUSE (Mobility Management Using Extension) una soluzione per la gestione della mobilità a livello applicativo basata sul protocollo SIP in grado di soddisfare la maggior parte dei requisiti richiesti ad un sistema di gestione della mobilità fra reti eterogenee. Tale soluzione è stata inoltre implementata, per testare sul campo le sue capacità e la sua interoperabilità con diversi terminali e Server SIP.

Inoltre, al fine di valutarne le prestazioni è stata effettuata una campagna di misure nel mondo reale utilizzando alcune reti di accesso commerciali (sia 3G che WiFi).

Infine, si propone una metodologia per la valutazione delle prestazioni delle differenti soluzioni di gestione della mobilità precedentemente illustrate. Tale metodologia si compone di un semplice modello analitico applicabile sia ad un caso ideale (senza perdite) che un caso reale con un predefinito tasso di perdita dei pacchetti. Il modello è stato applicato a tre soluzioni MMUSE, MIPv4 e SIP re-INVITE.

Acknowledgements

Sebbene tutta la tesi sia scritta in inglese, preferisco ringraziare le persone che mi hanno accompagnato in questi anni di dottorato nella mia lingua madre, l'italiano in quanto è la lingua con cui meglio riesco ad esprimere i miei sentimenti.

Non posso non partire con ringraziare la persona che mi ha introdotto in quel bellissimo mondo che è la ricerca universitaria ovvero il mio tutor e amico Stefano Salsano. Ricordo ancora il giorno in cui dopo timido e impacciato mi recai nel suo ufficio per chiedere la tesi, ma ricordo ancora meglio il giorno in cui lui mi propinò la possibilità di fare il dottorato. Accettai di getto e anche se da poco ho rinunciato alla ricerca, sono ancora convinto di aver fatto la scelta giusta.

Grazie Stefano, grazie per l'opportunità, grazie per esserci sempre stato disponibile per me nonostante i tuoi mille impegni giornalieri, per avermi fatto crescere professionalmente e per avermi, come hai detto tu stesso al piccolo Matteo "reso bravo".

Un grande grazie lo devo anche Gianluca Martiniello e a Luca Veltri che con Stefano hanno dato inizio a quello che poi è diventata la mia tesi di dottorato. Ho avuto l'occasione di lavorare con Gianluca solo un paio di mesi, e il mio rammarico è che sono stati troppo pochi. A Luca devo circa un milione e mezzo di consulenze telefoniche e via mail, è buffo pensare al fatto che ho avuto l'opportunità di scrivere molti articoli con lui, di contribuire alla mitica libreria mjsip ma che, di fatto, l'ho visto di persona una sola volta.

E come non ringraziare tutto il netgroup, questo stupendo gruppo di lavoro creato da Stefano, dal Prof. Blefari e dal Prof. Bianchi e che è stato il mio ambiente di lavoro in questi anni. Come dimenticarsi di tutti gli amici cui ho chiesto consulenze di inglese in cambio di consulenze su office, come dimenticare tutti i sorrisi di questi anni con Francesca Lo Piccolo, Dario, Lorenzo, Alessandro, Giovanni, Saverio, Arianna, Simone, Francesca Martire, Vito, Vincenzo, Mimmo. Grazie a tutti per aver arricchito scientificamente ed umanamente la mia vita.

Un grande grazie anche a Fabio Ricciato e Peter Reichl che hanno reso possibile il mio internship in FTW e grazie ad Alessandro, Marina, Danilo e Sabine per la loro squisita ospitalità e amicizia.

Infine un grazie particolare a quella che lo scorso giugno è diventata mia moglie Alessandra. Grazie per aver colmato la mia vita con la tua presenza e per essere sempre stata

al mio fianco in ogni circostanza. Grazie per aver accettato con serenità alcune scelte difficili. Ma soprattutto grazie per il figlio che porti in grembo, che renderà ancora più viva la nostra famiglia. E grazie anche a te piccoletto, grazie perché il tuo arrivo mi sta donando emozioni nuove, nuove consapevolezze e nuove sfide. Spero tanto di essere un buon padre per te e spero che tu, nella vita che ti accingi ad affrontare, possa incontrare persone stupende come quelle che sto ringraziando in questa pagina.

Contents

Mobility Management in Next Generation Networks

Introduction

Nowadays, the range of available wireless access network technologies includes cellular or wide-area wireless systems, such as cellular networks (GSM/GPRS/UMTS) or WiMax and local area or personal area wireless systems, comprising for example WLAN (802.11 a/b/g) and Bluetooth. A great part of today's mobile terminals are already capable of having more than one interface active at the same time. In addition, the heterogeneity of access technologies will likely increase in the future, making the seamless integration of the different ways in which a user can access the network a key challenge for Next Generation Networks. Services need to be provided to the user regardless of the particular access technology used, and IP protocol will be the common language for this integration at the network level.

The choice of the network interface to be used at a given time can be based on economical or performance considerations, anyway it is desirable that the user perceives the service in a seamless way, notwithstanding the changes of access interface (and technology). It is not easy to fulfill this requirement as moving across different access technologies may imply changes in the parameters of the communication, e.g., the IP address. The research community has given several answers to these needs, proposing different Mobility Management approaches, which can be classified according to the layer at which they operate. A first option is to work at the network layer, as Mobile IP does. Alternative approaches provide seamless service fruition by operating at the application layer; among them, a favorite choice is to rely on the SIP protocol for signaling purposes. Following this approach, in this thesis we propose a SIP-based solution that supports vertical handovers without disruption of real-time multimedia communication services. The solution is called “Mobility Management Using Sip Extension” (MMUSE).

This Thesis is organized as follows. In chapter 1 we introduce the reference scenario and an overview of the most important mobility solutions in literature. MMUSE, our solution is shown in chapter 2 introducing its main concepts and analyzing the details of the signaling procedures and handover policy. Chapter 2.6 analyzes MMUSE’s performance evaluation with respect to other handover solutions in literature. Finally chapter 4 shows a solution that improves the MMUSE architecture scalability.

1 Handover solutions overview

1.1 Reference Scenario and Requirements

Figure 1.1 shows a Mobile Terminal (MT) that can connect to different Access Networks (ANs) (AN1, AN2 and AN3). The different ANs could be based on different wireless or wired technologies (e.g. Wi-Fi, Bluetooth, GPRS/EDGE, 3G/HDSPA, WiMax, fixed Ethernet); the MT could be connected to more than one access network at the same time, if it has more than one physical network interface. Note that the ANs can provide public or private IP addresses to the MT (in most typical scenarios the Access Networks are likely to provide private IP addresses). For example, in Figure 1.1 AN1 and AN3 provide a private address (as shown by the NAT box), while AN2 provides a public address. The MT needs to be reachable in order to receive incoming calls on whatever network it is roaming. The MT wants to communicate with a "Correspondent Terminal" (CT), which can have a public address (like CT2 in the figure) or a private IP address (like CT1 in the figure). When a communication session is active, the MT may need to change the AN, as the interface that is using may become not available due to loss of signal, or it could suffer of a high packet loss or packet delay.

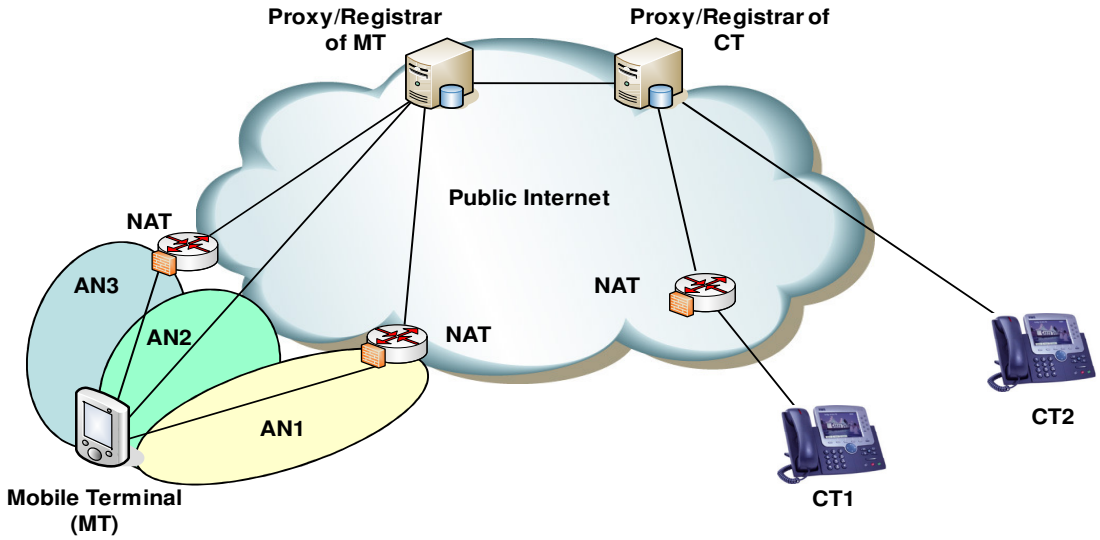


Figure 1.1: Reference Scenario

Our analysis focuses on "vertical" handover, between heterogeneous networks; the "intra-technology" and "intra-network" handovers are out of the scope of our activity. For

example, a MT connected to a cellular network may perform several handovers among different cells within the cellular network, but it will remain attached to the same AN, with reference to the scenario of Figure 1.1. “Vertical” handovers are defined as the switch between two different access technologies or the switch between two different access networks operating with the same technology, if the IP address provided to the MT changes as a consequence of the handover.

The focus of this thesis is only on Mobility Management for real time services that use UDP as transport protocol: Mobility Management and seamless handover for services that use TCP transport is out of our scope.

The Mobility Management procedures basically consist in:

- allowing users to be reached on whatever access network they are;
- allowing the “handover” of active real-time communication session from an access network to another one.

Hereafter we identify the requirements of an “optimal” Mobility Management solution:

1) The vertical handover must be as fast as possible. This means that the user should not perceive any service interruption. If it is not possible to completely hide the effect of the handover, then the service disruption should be minimized.
2) When switching from an access network to another, the Mobility Management signaling should be sent over the new target network, since the old one could suddenly become unavailable; in such a case it is necessary to perform the whole handover procedure on the new network (this is known as “Forward” handover). On the contrary, if the old network is still available, the availability of both networks can be exploited in order to assist and speed up the whole procedure.
3) The Mobility Management solution should be compatible with NAT traversal. Users should be able to roam from an access network to another, even when one or both access networks use private IP addressing and lie behind a NAT.
4) The Mobility Management solution should not require modifications to the CTs. All existing terminals should be able to interoperate with roaming MTs.
5) Existing User Agents (UAs) in the MT should not be modified in order to be able to exploit the roaming capability provided by the Mobility Management solution.
6) The Mobility Management solution should not require additional support in the access networks. The access networks are only required to provide IP connectivity.

7) The capability to offer Mobility Management services should not be an exclusive prerogative of the network operators.
8) The actual location and the movements of the user should not be visible from the CT, in order to preserve the privacy of the users.
9) The Mobility Management solution should properly interact with the User Registration procedures and with existing solutions for handling Personal Mobility (for example these solutions allow a user to use a set of mobile and fixed terminals in parallel or in sequence, as he desires). In other words, the Mobility Management solution should be able to complement current services offered by existing SIP proxy/registrar servers, without the need of redesigning the service logic or modifying the SIP protocol implementation in these servers.

1.2 Solutions for Mobility Management

Focusing on IP based devices and services, these requirements for mobility management solutions can be tackled at various levels of the protocol stack from application level (e.g. SIP based solutions) to network level (e.g. Mobile IPv4, Mobile IPv6) to link layer level (e.g. 802.21). A large number of different mobility management solutions operating at these different levels have been proposed so far, both in the literature and in the standard bodies.

This section shows a summary of these mobility solutions.

1.2.1 Link Layer

1.2.1.1 802.21

802.21 [10] is a standard defined by the IEEE in order to optimize handovers among heterogeneous IEEE 802 networks and to facilitate handovers between IEEE 802 networks and cellular networks.

This standard provides link layer intelligence and other related network information to upper layers to optimize handovers between heterogeneous networks. This includes media types specified by Third Generation (3G) Partnership Project (3GPP), 3G Partnership Project 2 (3GPP2), and both wired and wireless media in the IEEE 802 family of standards. In this standard, unless otherwise noted, media refers to method/mode of accessing a

telecommunication system (e.g., cable, radio, satellite), as opposed to sensory aspects of communication (e.g., audio, video).

The purpose of this standard is to enhance the experience of mobile users by facilitating handovers between heterogeneous networks.

This standard supports cooperative use of information available at the mobile node and within the network infrastructure. The mobile node is well-placed to detect available networks. The network infrastructure is well-suited to store overall network information, such as neighborhood cell lists, location of mobile nodes, and higher layer service availability. Both the mobile node and the network make decisions about connectivity. In general, both the mobile node and the network points of attachment (such as base stations and access points) can be multi-modal (i.e., capable of supporting multiple radio standards and simultaneously supporting connections on more than one radio interface).

The overall network can include a mixture of cells of drastically different sizes, such as those from IEEE 802.15, IEEE 802.11, IEEE 802.16, 3GPP, and 3GPP2, with overlapping coverage. The handover process can be initiated by measurement reports and triggers supplied by the link layers on the mobile node. The measurement reports can include metrics such as signal quality, synchronization time differences, and transmission error rates. Specifically the standard consists of the following elements:

a) A framework that enables service continuity while a mobile node (MN) transitions between heterogeneous link-layer technologies. The framework relies on the presence of a mobility management protocol stack within the network elements that support the handover. The framework presents media independent handover (MIH) reference models for different link layer technologies.
b) A set of handover-enabling functions within the protocol stacks of the network elements and a new entity created therein called the MIH Function (MIHF).
c) A media independent handover Service Access Point (called the MIH_SAP) and associated primitives are defined to provide MIH Users with access to the services of the MIHF. The MIHF provides the following services:
    i. The Media Independent Event service that detects changes in link layer properties and initiates appropriate events (triggers) from both local and remote interfaces.

ii. The Media Independent Command service provides a set of commands for the MIH Users to control link properties that are relevant to handover and switch between links if required.

iii. The Media Independent Information service provides the information about different networks and their services thus enabling more effective handover decision to be made across heterogeneous networks.

iv. The definition of new link layer service access points (SAPs) and associated primitives for each link-layer technology. The new primitives help the MIHF collect link information and control link behavior during handovers. If applicable, the new SAPs are recommended as amendments to the standards for the respective link-layer technology

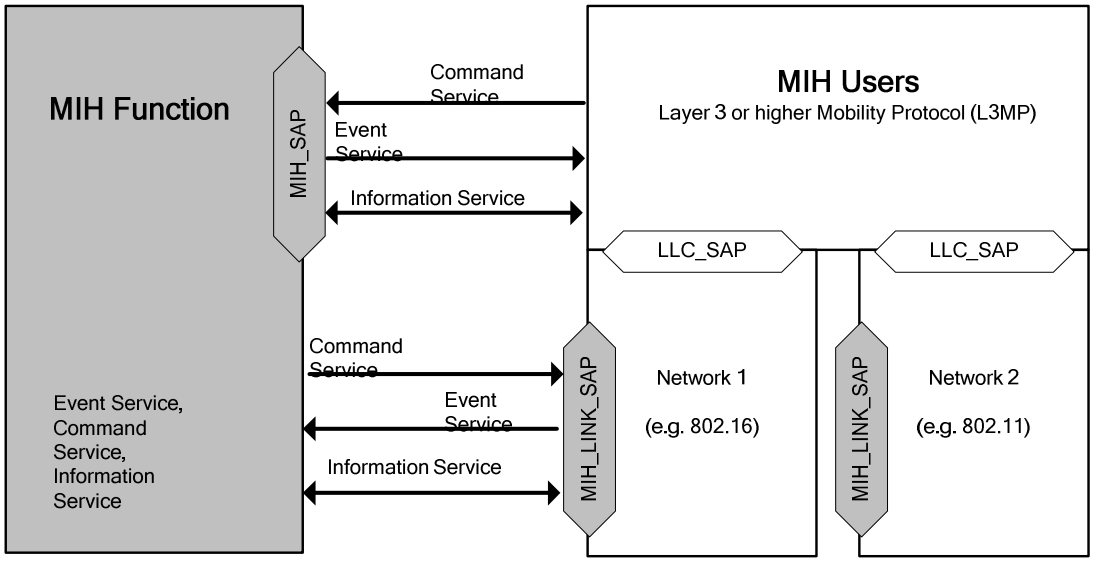


Figure 1.2: MIH services and their initiation

Figure 1.2 shows the placement of the MIHF within the protocol stack of a multiple interfaced MN or network entity. The MIHF provides services to the MIH Users through a single media independent interface (the MIH service access point) and obtains services from the lower layers through a variety of media dependent interfaces (media-specific SAPs).

1.2.2 Network Layer

1.2.2.1 Mobile IP

Mobile IP (MIP) [9] is a mobility solution working at the network layer. IPv4 assumes that every node has its own IP address, which should remain unchanged during a communication session. MIP introduces the concepts of Home Address (the permanent address of the MT) and of Care-of-Address (CoA). The latter is a temporary address assigned to the MT as soon as it moves from its home network to a foreign one. A specific router in the home network (Home Agent) is informed as soon as the node acquires the CoA in the foreign network (from a so-called Foreign Agent). The Home Agent acts as an anchor point, relaying the packets addressed to the Home Address towards the actual location of the MT, at the Care-of-Address.

Using Mobile IP for real-time communications has some drawbacks. A first well-known problem is triangular routing, i.e., the fact that the packets sent to the MT are captured by the Home Agent and tunneled, while the MT can send packets directly to the CT. This asymmetric routing adds delay to the traffic towards the MT, and delay is an important issue in VoIP. The fact that the packets are tunneled also means that an overhead of typically 20 bytes, due to the IP encapsulation, will be added to each packet. Still another drawback of using Mobile IP is that each MT needs a permanent home IP address, which can be a problem because of the limited number of IP addresses in IPv4.

A number of works have built upon MIP in order to overcome its drawbacks. A notable one is Cellular IP [12] that improves MIP providing fast handoff control and paging functionality comparable to those of Cellular networks. Being a network level solution, Cellular IP requires support from the access networks and it is suitable for “micro-mobility” i.e. mobility within the environment of a single provider.

As for the triangular routing problem in Mobile IP, there was a tentative to solve it by sending binding updates to the CT in order to inform this node about the actual location of the MT. Unfortunately this has become a standard only for Mobile IPv6 (MIPv6) while it has not been adopted for MIPv4. There are also other MIPv6 extensions that try to improve Mobile IP operation in terms of handover speed such as Hierarchical Mobile IP and Fast Handovers.

With Hierarchical Mobile IPv6, a new node, called Mobility Anchor Point (MAP) is introduced and located close to the access network. This can substantially speed-up the binding procedure and reduce the overall handover time. Furthermore, Hierarchical Mobile IPv6 allows MTs to hide their location from CTs when using Mobile IPv6 route optimization. Fast Handovers for MIPv6 is a mechanism that tries to minimize communication latency by allowing the MT to send and receive packets as soon as it detects a new access network. The main drawbacks of Mobile IPv6 and of its enhancements are that they require IPv6 to be deployed in terminals and in the network, and (as IPv4) they rely on the support from network devices in each access network to work properly.

1.2.3 Transport layer

1.2.3.1 Mobile SCP

Stream Control Transmission Protocol (SCTP) [RFC4960] is an end-to-end transport layer protocol. The main SCTP advantage is the native multi-homing support.

Multi-homing is the ability for a single SCTP endpoint to support multiple IP addresses. In SCTP this feature is typically user for keep alive a session in the presence of network failures.

An SCTP transport address is a pair of an IP-address and a port number as in the case of TCP. But an SCTP endpoint can be identified by a sequence of SCTP transport addresses all sharing the same port number.

An association is a connection between two SCTP endpoints.

An SCTP endpoint can use multiple IP-addresses for an association. These are exchanged during the initiation of the association. The multiple addresses of the peer are considered as different paths towards that peer. This means that a server must use multiple IP-addresses to provide the mobile client with multiple paths. These will be used while moving between locations.

It should be mentioned that this path-concept is used only for redundancy, not for load sharing. Therefore one path is used for normal transmission of user data. It is called the primary path.

Also the SCTP extension described in [RFC5061], which is called ADDIP, include the capability of dynamic IP address reconfiguration during an association. This means that it

allows an SCTP endpoint to add a new IP address, to delete an unnecessary IP address and to change the primary IP address used for the association.

Furthermore it is possible for an SCTP endpoint to signal to its peer which IP-address it should use as the primary path. This is very useful in case of multiple Internet accesses with different parameters. The combination of SCTP and its extension ADDIP is called “mobile SCTP” defined in [11].

1.2.3.2 HIP: Host Identity Protocol

Host Identity Protocol (HIP) architecture - RFC 4423 - defines a new global Internet namespace which uses public/private keys, instead of IP addresses, as Host Identifiers. HIP provides a solution to the flexibility limitation of the current Internet architecture by decoupling the double role of both topological locator and network interface identifier currently filled by IP addresses.

In the HIP architecture a Host Identity layer is introduced between the Transport layer and the Network layer so that a different binding of transport layer protocols is provided (Figure 1.3). The transport-layer associations, i.e., TCP connections and UDP associations are no longer bound to IP addresses but to Host Identities. In this architecture the end-point names and locators are separated from each other. IP addresses continue to act as locators while the Host Identifiers take the role of end-point identifiers. Host Identifiers are locally mapped to the corresponding IP addresses and are used as source and destination addresses in the IP packets. In other words, HIP aware applications can bind their sockets directly to a <Host Identifier, Port> pair and they never see the actual remote IP address.

Since Host Identities can be bound to different IP addresses, with HIP, all existing transport associations can be easily moved to one address to another if one address becomes unreachable or if a new address become available. Consequently, HIP provides for interworking mobility and multi-homing without requiring additional mobility management signaling.

Furthermore, HIP provides for process migration and clustered servers. Indeed, it is possible to move all the active transport associations if a Host Identity is moved from one physical computer to another. Similarly, with HIP it is possible to distribute a Host Identity over different physical machines to provide clustered-based service without changing the client end-point.

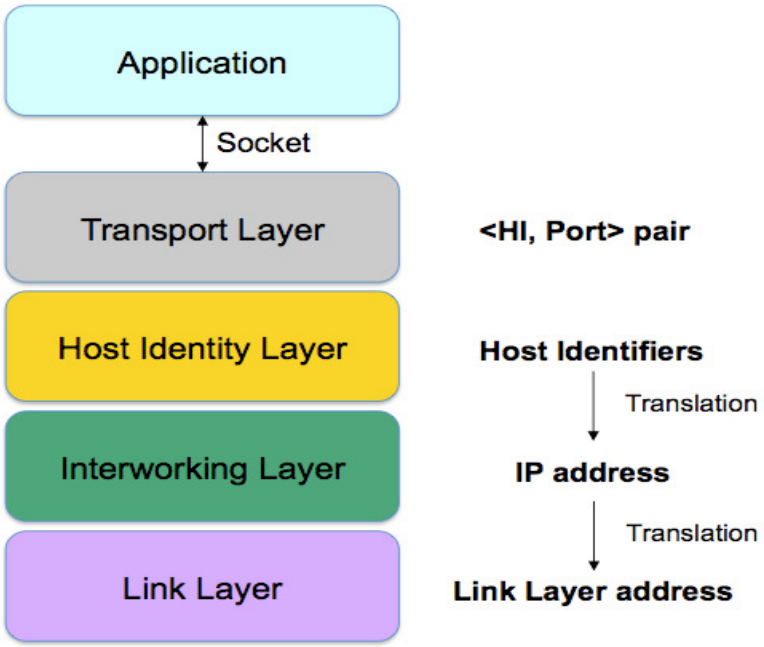


Figure 1.3: HIP Stack architecture

1.2.4 Application layer

1.2.4.1 SIP re Invite

According to the SIP protocol, an INVITE message is sent by a terminal to its correspondent to setup a communication session. The traditional SIP mechanism to provide terminal mobility during an active session [13] involves not only the MT but also the CT, which is the other party engaged in the call. It foresees that the MT sends another INVITE message to the CT to communicate the information about the new parameters of the communication session after the handover. Although this solution solves the problems of Mobile IP, it has some drawbacks too. The second INVITE (commonly referred to as "Re-INVITE") is sent end-to-end, and this could lead to high delays. Moreover, the handover procedure relies on the capability of the CT to handle this procedure.

Other SIP-based approaches have been proposed to manage mobility, addressing the shortcomings of the end-to-end "Re-INVITE" mechanism. For example, in [14] it is assumed that the MT connects to the Internet through different ANs and that each of them has its own so-called "Base Station". The Base Stations are able to handle the vertical mobility and perform a handover, by moving a communication from a base station to the other. Actually, the handover procedure is split in two phases. First the MT contacts the old Base Station and

asks to receive/send packets over the new AN, using an INVITE message, which makes use of the JOIN SIP header [15]. For a certain time, media packets will be duplicated and sent over both wireless networks. As soon as the packets reach the MT through the newly activated interface, a re-INVITE message is sent by the MT to the CT through the new Base Station. Then, the media will flow through the new Base Station and over the new Access Network; a SIP BYE message will be sent to close the session with the old Base Station and a REGISTER message will be sent to the user "home" Registrar server to update the user contact information. This solution improves the performance of traditional SIP mechanisms in terms of handover duration and packet loss, but it still requires the involvement of the CT. Also, it requires a complex sequence of SIP transactions (INVITE, "Re-INVITE", BYE and REGISTER) for each handover and it does not address NAT traversal issues.

In the following section we will see how our proposed SIP based Mobility Management solution tries to overcome these limitations.

2 MMUSE: Mobility Management Using SIP Extensions

In this Chapter we present the main features of MMUSE (Mobility Management Using Sip Extensions). MMUSE is our Mobility Management solution based on ad-hoc extensions of SIP protocol in order to support mobility. We presented this extensions in at the IETF (Internet E Task Force) in two drafts [4] and [5] in order to standardize its.

2.1 MMUSE overview

Let us consider the reference scenario depicted in Figure 1.1. A MT is equipped with multiple network interfaces; each of them is assigned and uses a different IP address, when connected to different Access Networks. The MT uses the SIP protocol for the setup of multimedia sessions. Our aim is to allow MTs to move among access networks (both wireless and wired) taking into due account the requirements listed in Section 1.1.

We focus our attention on a scenario including a so-called Session Border Controller (SBC). A SBC is a device typically located at the border of an IP network which manages all the sessions for that network. An SBC may perform several functions; for example it can provide NAT traversal features and privacy for the users of the internal network, by hiding the network structure behind it. It is an important component of several VoIP solutions. The SBC can be used by an enterprise, to allow its hosts located in a private network to make and receive calls or can be used by a public VoIP provider to offer VoIP services to enterprises.

Our basic idea is to extend the signaling and media functionalities of the SBC in order to manage mobility. To this aim we introduce a new entity, called Mobility Management Server (MMS), within the SBC (MMS functionality are shown in 2.2.2). We also assume that the MMS cooperates with another entity that we introduce within the MT, called Mobility Management Client (MMC) (see 2.2.1). Both the SIP User Agent (UA) on the MT and the one on the CT remain unaware of all handover procedures, which are handled by the MMC and the MMS. On the MT, the UA just sees the MMC as its outbound proxy and forwards the normal SIP signaling and media flows to it; the MMC relays them to the MMS/SBC; from there on they follow the path determined by the usual SIP routing procedures. The MMS/SBC is a permanent anchor point both for signaling and media; we note that its presence is needed in any case to allow NATed UAs to be reachable. Figure 2.1 shows the architecture of the proposed solution, where the SBC is able to act as “meeting point” between the CT and

the MT, independently from the Access Network(s) on which the MT is located. For the sake of simplicity, in Figure 2.1 we only show a single "centralized" SBC/MMS, while a real world solution would take into account the scalability issues. A set of coordinated SBC/MMS needs to be deployed to cover the needs of a large number of mobile users.

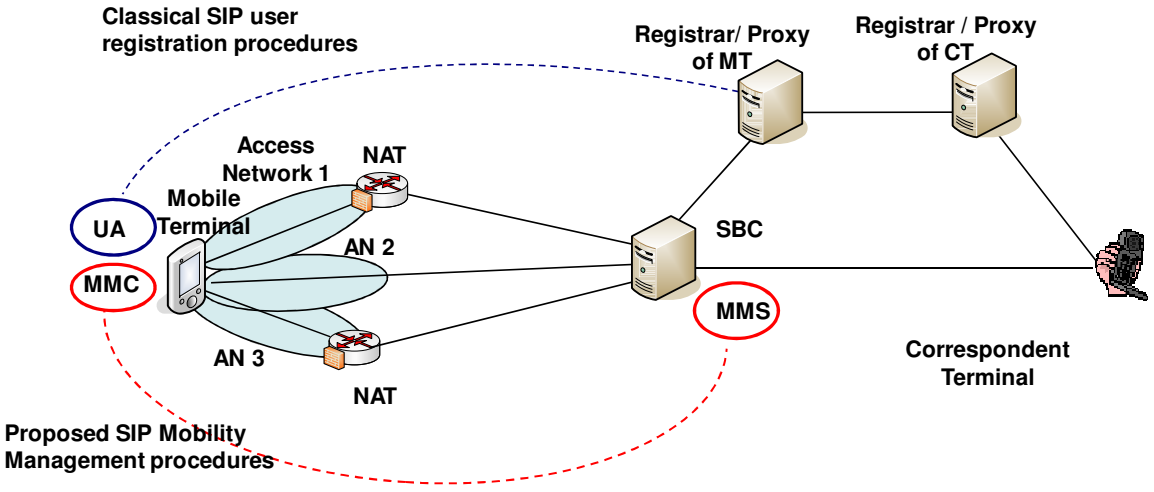


Figure 2.1: Architecture of the proposed solution

We have also defined specific signaling procedures (detailed in section 2.3), exchanged between the MMC in the MT and the MMS, so that the latter entity is always informed about the actual location of the MT. In particular, every time that the MT moves across two access networks, a location update SIP message is sent to the MMS, and this is done over the new network, to make it possible to complete the procedure even if the old network is suddenly not available (see Figure 2.2-A). If the MMS receives a call addressed to one of its served MT, it will forward it to the correct interface, thanks to the state information that it keeps.

When the MT needs to change access network while it is engaged in a call, the procedure is almost identical, with the difference that in this case the MMC sends to the MMS a SIP message which contains the additional information required to identify the call to be shifted to the new interface (see Figure 2.2-D).

To minimize the duration of the handover, we duplicate the RTP flow coming from the MT during the handover, by using the MMC. When the MMC starts the handover procedures, it sends the handover request (a SIP REGISTER) to the MMS and, at the same time, it starts duplicating the RTP packets over both interfaces. In this way, as soon as the MMS gets the

handover message, the packets coming from the new interface are already available. The MMS can perform the switching in the fastest possible way and then send the reply back to the MMC (a SIP 200 OK). When the MMC receives the reply message, it stops duplicating the packets. We modified the time parameters of the retransmission procedure of the handover REGISTER message so that a fixed retransmission interval of 0.2s is used, thus minimizing the duration of the handover, even when signaling packets are lost.

As regards the regular SIP transactions, such as the establishment of a new session (see Figure 2.2-C), the termination of an existing session, or the registration with a specific SIP proxy (see Figure 2.2-B), they are kept unchanged, under the constraint that all SIP signaling has to pass through the MMC and MMS. These two entities modify SIP messages in order to mask the current position of the user and to direct both SIP signaling and RTP packets to the correct location, taking care of any NAT device in the middle of the path. As far as the SIP signaling is concerned, the MMS/SBC acts as a standard SIP proxy with additional functionalities. Instead, as regards the media, it behaves as a SIP Back-to-Back User Agent (B2BUA): it divides the media path in two parts and interconnects them; each UA is led to believe that the other party is located at the MMS/SBC address.

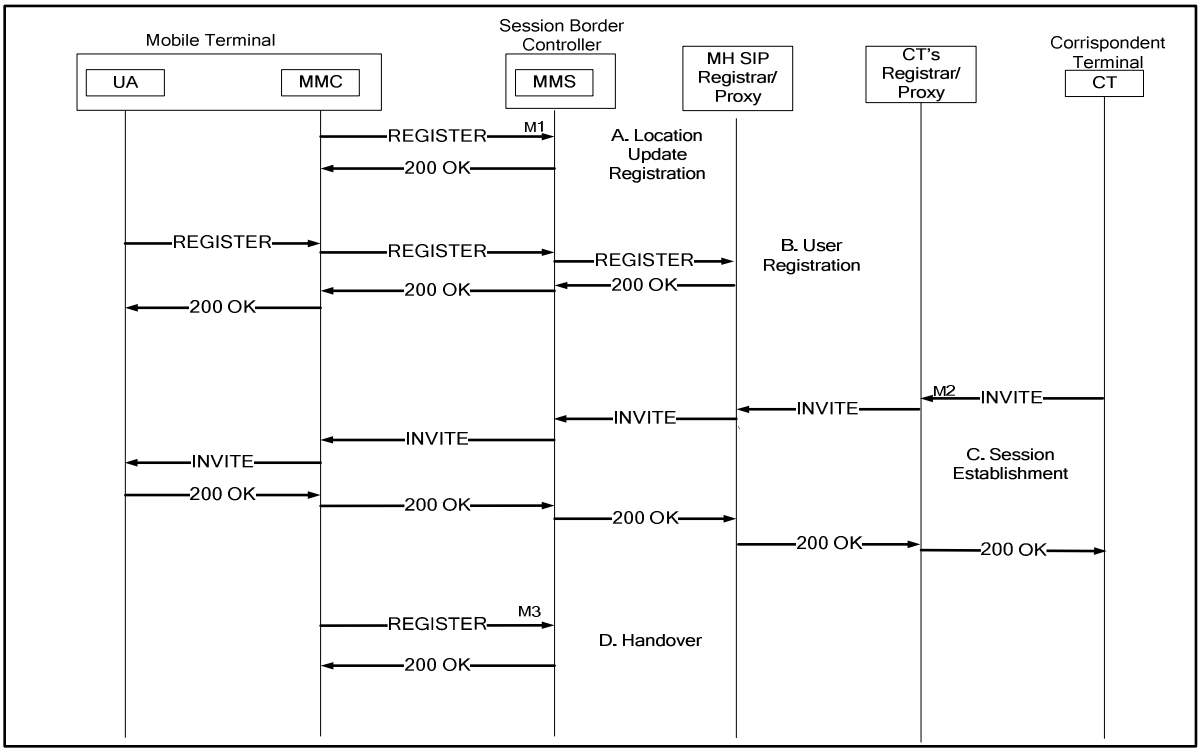


Figure 2.2: MMUSE signaling procedures

In MMUSE, the user location mechanism is split in two levels. The MMS/SBC has a complementary role with respect to the SIP Registrar/Proxy server of the traditional SIP architecture. The SIP protocol foresees that the SIP Registrar receives information about the location of the user, acquired during the registration procedure, so that the "inbound" Proxy server of the user can forward incoming requests to the user. Nevertheless, in our case this location changes every time that the MT switches to a different network. To provide a more flexible solution, we let the MMS control the movements of the MT among different access networks through an internal registration, and leave the Registrar/Proxy unaware of such movements, by using a contact information that always points to the MMS itself.

In this line of reasoning, we distinguish between the identification of the user and that of the MT. In the classical SIP architecture there is no such distinction: a user can have more than one terminal (i.e., SIP User Agent) active at the same time, at different locations, and all terminals are seen as "contact addresses" of the user. When a terminal changes its IP address, a registration is sent to the Registrar, which will simply change the "contact address" for the user. In our solution, as stated above, we want to separate the user level registration from the management of the MT mobility. Therefore, we need to explicitly identify the MT by introducing an identifier, called terminal ID. This identifier is representative of a MT and it is used by the MMS to identify the MT and to keep track of its location (IP address). This identifier does not need to be understood outside the context of the MMC-MMS relationship: the user SIP Registrar will receive a registration in which the "contact address" points to the MMS.

We developed the MMC as a separate entity on the mobile terminal, making it able to interact with the UA via SIP messages. The MMC communicates with the operating system; thus, it is aware of the interfaces that are active at a given time. The MMC has the task to select the preferred interface and to change it, if needed. The main advantage of this approach is that it does not require any modification to the UA.

Section 2.5 shows the possible handover criteria used in MMUSE. We have developed tree types of handover criteria:

1) Based on signal's power (as shown in 2.5.1)
2) Based on quality of link (as shown in 2.5.2)
3) Based on location's information (as shown in 0)

The proposed architecture, based on SBC/MMS, may suffer of scalability problems, especially because an SBC/MMS needs to have the media relay functionality for all MTs under its control. We are currently working on how to distribute the SBC/MMS functionality to address scalability issues, a possible solution of this problem is shown in section 4. Also, we observe that this problem is common to all other solutions that are able to provide NAT traversal also for symmetric NAT, (e.g. STUN-Relay/TURN): they need to introduce a media relay element. Moreover, adding a media relay on the path is needed anyway when it is necessary to provide lawful interception services.

2.2 MMUSE Architecture

The "traditional" way to perform application level mobility with SIP is using the "Re-Invite" mechanism. It requires that the correspondent terminal is able to perform the handover, under the control of the mobile terminal. This approach does not match very well with the requirements listed above, as: it does not care much about performance (the procedure is end-to-end); NAT traversal issues are not taken into account, it relies on the capability of the remote terminal, it does not provide any privacy with respect to user movements, soft handover is not considered. For these reasons we preferred to introduce the element called "Mobility Management Server" to handle the terminal mobility across different access networks and to assist in performing the handover procedure.

The fundamental concepts of the proposed solution can be illustrated with the help of Figure 2.1. Mobile Terminals (MTs) have access to different networks like WiFi, Cellular 3G (in the figure, AN1, AN2 and AN3 network), which can overlap their coverage areas. The MT has separate interfaces, each one dynamically receives its (private or public) IP address from the corresponding wireless network. Multiple network interfaces in the MT can be active at the same time making it possible to realize a "soft handover". The MT logically contains the User Agent (UA, i.e. the SIP client) and a Mobility Management Client (MMC). The MT uses a Session Border Controller (SBC) to access VoIP services from IP access networks often based on a private IP addressing scheme and operating behind a NAT/FW box. The SBC contains a Mobility Management Server (MMS) which is the main entity controlling the user mobility. Thanks to the interaction between the MMC in the mobile terminal and the MMS in the SBC the device can move between IP subnets, allowing the UA to be reachable for incoming requests and to maintain VoIP sessions across subnet changes. The "CT" node shown in the picture is the Correspondent Terminal that communicates with the MT. SIP Registrar functionality that are not directly related to handover/mobility management procedures can be performed by an external SIP Registrar/Proxy, as shown in the figure. Obviously the MMS and SIP Registrar can be implemented in a single element if required.

2.2.1 The Mobility Management Client

The Mobility Management Client is implemented as shown in Figure 2.3 as a separate entity running on the MT. The MMC hides all mobility and NAT traversal functionality to the

terminal SIP User Agent by relaying both signaling and media flows. In this case the SIP User Agent sees the MMC as default "outbound proxy" (which means that the UA will send all SIP message to the MMC) and it has no knowledge of the handovers. Existing SIP UAs can be easily supported/reused without any changes.

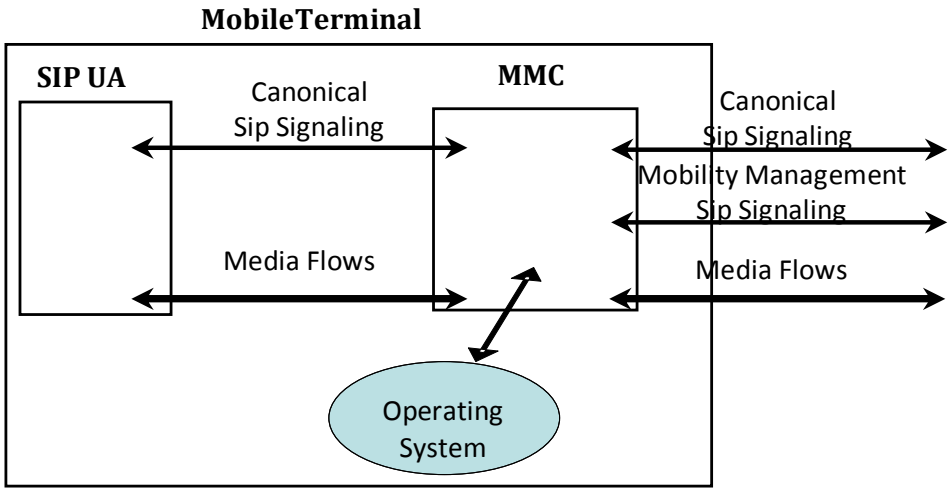


Figure 2.3: The Mobile Terminal

In order to configure the IP addresses on the MT interfaces, existing mechanisms are used (e.g. PPP on the 3G interface, DHCP on WiFi LAN and Bluetooth PAN). When multiple interfaces are active, the MMC needs to select the preferred interface for sending/receiving the media flows (while the terminal is involved in a call) or for exchanging SIP signaling (both during calls and in idle state). The choice of the selected interface performed by MMC may depend on cost aspects and/or on QoS issues (signal strength, perceived packet loss and/or delay).

2.2.2 The Mobility Management Server

A SIP Session Border Controller is a session-aware device that manages SIP calls at the border of an IP network. It is aware of both signalling and media flows. An SBC may have several functions, one of the most interesting is solving the problem of NAT/firewall traversal, dealing with different NAT and/or UA behaviors. In this respect, an SBC provides NAT/firewall traversal without additional customer premise equipment, and without the replacement of existing firewalls and NATs. An SBC does not require any additional STUN/TURN node nor STUN/TURN protocol support, neither at UA nor at SBC side. Besides NAT traversal, the SBC may have several function, we will only list two of them: (i) the SBC can provide media interworking function for different media-related functionalities such as:

media transcoder, media encryption and protection against various media-based attacks; (ii) the SBC can provide signaling and media wiretapping system, which can be used to enforce requests for the lawful interception of communication sessions.

In order to manage the user mobility, we propose to add the MMS element into the SBC. The MMS is an “anchor point” for the media flows which are transmitted over the wireless access networks directed to (and coming from) the MT. When the MMC in the MT detects that a handover is needed, it will request the handover to the MMS (via a SIP message) over the “target” network. Then the MMS (in the SBC) will update its media proxy and will start transmitting and receiving the media over the target network (details are provided in the next section). Note that the entire handover procedure is handled by the MT and the SBC, letting the Correspondent Terminal (CT) (and other SIP intermediate nodes) completely unaware of what is occurring.

The SBC enhanced with the MMS is needed to manage MT handoffs between different access networks providing service continuity and NAT traversal. The SBC is able to process both SIP protocol header fields and Session Description Protocol (SDP) bodies in order to force itself as relay for the media packets.

2.3 MMUSE signaling aspects

This section describes the specific procedures for mobility management (off-call and on-call) and shows how the canonical SIP procedures (user registration and session establishment) are realized under the proposed solution. The procedure for mobility management related to the off-call part is meant to be used by the MT (UA+MMC) to tell the MMS about its current selected/active interface (or to communicate changes to such selection) when not in a call, for this we use regular REGISTER sent from the MMC to the MMS. The on-call mobility management related to the on-call part is meant to be used by the MT (UA+MMC) to tell the MMS about a change in its current selected/active interface when in a call, for this we use regular REGISTER sent from the MMC to the MMS with an additional "Handover" header which contains the reference to the active session(s) to which the handover is referred. In both cases an additional parameter to be added to the "Via" header for correct routing of responses to the MMC is needed and used.

2.3.1 Role of the Session Border Controller

From the point of view of SIP signaling, an SBC can act as a SIP B2BUA (Back-to-Back UA) or as a special SIP proxy. In the former case, the SBC works as an intermediate node that breaks the signaling path between two UAs and interconnects them (e.g. setting up a call) by means of establishing separate end-to-end connections between itself and each remote UA. In the latter case the SBC does not break the signaling path between the two UAs; instead it relays signaling requests and responses between remote UAs and other proxies, operating all SBC-specific function extending the normal proxy behavior as defined by RFC 2361. In addition to what is defined in the SIP standard for the operation of a SIP proxy, the SBC will modify the description of media session contained in the SDP, and some other SIP header fields like for example the Contact header field. Despite this extended behavior, obviously all outgoing signaling remains fully compliant with SIP standards. In our solution, we preferred to use a “proxy like” SBC as it is lighter and more “transparent” with respect to the SIP signaling among the endpoint UAs. The MMS/SBC behaves like a stateful proxy and it will receive ALL incoming and outgoing SIP messages to and from the mobile terminal. In particular it will process the incoming and outgoing INVITE messages used for setting up the calls.

2.3.2 Choice of SIP messages for Mobility Management

The procedures that need to be defined for mobility management can be classified as: out-of-call mobility management, call setup, in-call mobility management (i.e. handover).

The "traditional" SIP based mobility management foresees REGISTER messages for out-of-call mobility management, INVITE messages for call setup and RE-INVITE messages for in-call mobility management (i.e. handover). In our proposal we use REGISTER messages for both out-of-call and in-call mobility management and INVITE messages for call setup. There are two types of REGISTER message, one is related to user registration towards the user Registrar server, the second type of REGISTER is a "mobility" registration between the MMC and the MMS. The second type of REGISTER is used both in case of out-of-call mobility management and in case of a handover during an active call.

The user registration REGISTER messages and the mobility management REGISTER message should be distinguishable. In the implemented solution they are simply distinguished by the SIP destination (i.e. the SIP Registrar or the MMS. Moreover the in-call mobility management REGISTER also carries a reference to the active call which is under handover (namely the call reference).

2.3.3 Choice of terminal identifiers for SIP mobility management procedure

By means of the above described mobility management REGISTER messages, the SBC/MMS becomes aware of the current position of the MT, and can correctly route any new request or response messages addressed to the mobile UA. A key aspect concerning this procedure and its usage is the UA identification and addressing. In general a user may have more than one UA active, each one attached to a network with its own IP address (and SIP port). When sending a request or receiving a response, SIP usually identifies the users through the URLs present within the From and To header fields and through the request URI, while the actual address of the UA is normally present in the Via and Contact header fields. Unfortunately, neither the user URL nor the UA address can be used for UA identification since the former is not bound to a specific UA (more user's UAs can be present) while the latter changes each time the UA moves from one network to another and, in presence of NATs, it is not unique due to the normal reuse of private addresses. For this reason a proper UA identification mechanism would be needed, but current SIP standard does not provide such mechanism. We used an identifier that the MMC inserts as an additional parameter in

the Via header, and it is denoted as Mobility Management ID (section 2.3.4.2). This information is processed by the MMS and needs to be understood only by the MMS itself. The CT (and the SIP registrar/proxy where needed) will handle this information in a completely transparent way according to standard SIP signaling. The details of the solution can be found in the next sections, which reports the complete SIP messages related to the various procedures.

2.3.4 Routing of requests and responses

In this section we detail how SIP messages are routed among the different entities. The challenge is to deliver incoming SIP requests and SIP responses to the MT, notwithstanding its mobility.

As for incoming SIP requests, when the UA in the MT performs the User Registration procedure (Section 0) the MMS rewrites the Contact header filed so that it points to the MMS itself. Therefore incoming requests for the MT will be forwarded by the SIP incoming proxy to the MMS. Similarly when outgoing requests are sent from the MT to a Correspondent Terminal the MMS will rewrite the Contact header so that the CT will consider the MMS as the destination of future requests to the MT. When incoming requests arrive to the MMS, the MMS will forward them to the current IP address of the MT, as updated by the MT using the Location Update procedure (Section 2.4.1). As for responses to outgoing SIP requests sent by the MT, the MMS adds a new parameter in the Via header field (see Section 2.3.4.2). This parameter is used by the MMS itself to route the response.

2.3.4.1 Use of Contact header field

The solution foresees that the MMS rewrites the Contact header when forwarding outgoing SIP requests coming from the MT. The rewritten contact address will have as host part the IP address (or domain name) of the MMS, and as username part a hint to the original contact address. This rewritten Contact header is compatible with SIP specifications, and it is reversible from the MMS. For example, assume that user's AoR is user@domain.com and the original Contact header inserted by the UA is:

Contact: <sip:user@x.y.w.z:5080>

where x.y.w.z is the current IP address of the MT. The MMS rewrites the contact as follows:

Contact: <sip:/TOKEN-user/AT-x.y.w.z/PORT-5080@MMS_x.y.w.z>

where MMS_x.y.w.z is the IP address of the MMS, "TOKEN-" is a string that can be set by the MMS and "/" is used as escape character. When receiving an incoming request with the request URI corresponding to the above contact, the MMS will extract the original contact address user@x.y.w.z:5080 and will forward the message according to the information contained in its MMS mobility database.

2.3.4.2 MMID parameter in Via header field

According to SIP specification, each node in the request delivery chain adds a Via header field with its own IP address when forwarding the request, in order to be included in the response delivery chain. We propose that the MMC adds an additional parameter to the Via header. This parameter is called MMID (Mobility Management IDentifier) and it carries the identifier of the MT. The MMID parameter is used as an indication that the originator of the request is a mobile terminal and it could change its IP address even during the transaction. Therefore the value of the MMID will be used as a key into the MMS mobility database, in order to find the current IP address to send the response. With this mechanism, the solution is able to deal in a seamless way with an handover performed during a session establishment. As an example, a Via header sent by the MMC to the MMS may look like the following:

Via: SIP/2.0/UDP x.y.w.z;branch=z9h;MMID=user@domain.com

2.4 MMUSE: SIP Procedures

As described in the previous section, the mobility management involves four main functional entities. On the MT sides there are the SIP UA and the MMC, while on the network side there are the SBC with the MMS and a SIP Registrar.

In this section we show the Sip Messages exchange among the various entities during the standard sip procedures *(User Registration*, *Incoming* and *Outgoing Call*) and during the specific mobility management procedures (*Handover* and *Location Update* procedure).

Figure 2.4 provides a physical representation of the scenario that will be used to describe the signaling flows and in Table 2.1 we show the IP addresses and the UDP port used from various Sip entities.

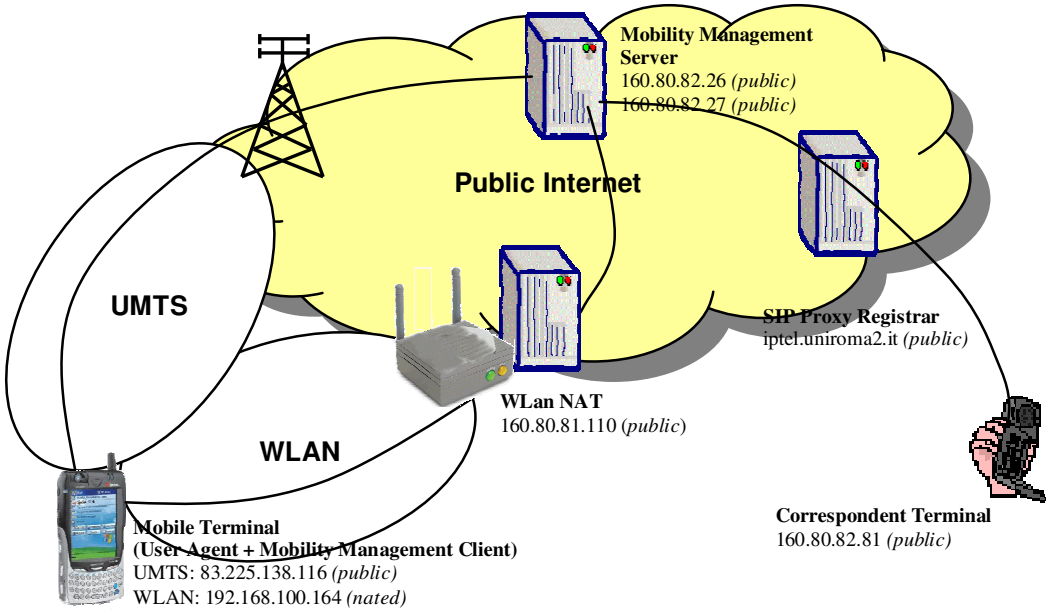


Figure 2.4: Layout of the demonstrator

Sip Entities	IP addresses: UDP Ports
User Agent (MT)	127.0.0.1:5050
Mobility Management Client	127.0.0.1:5060 (internal) 192.168.100.164:5070 (wifi) 83.225.138.116:5070 (umts)
Mobility Management Server	160.80.82.26:5070 (correspondent wifi) 160.80.82.27:5070 (correspondent umts)
Proxy Registrar	iptel.uniroma2.it:5061
User Agent (CT)	160.80.82.81:5060

Table 2.1: Addresses of the SIP Entities

2.4.1 Location Update Registration: initial and "off-call" mobility management

The Location Update Registration is the basic mobility procedure that allows a MT to notify the MMS about its "position" (or better its IP address) and select the currently preferred interface for sending/receiving SIP signaling and media flows. The sequence diagram of this procedure is shown in Figure 2.5. The MMC in the MT sends a Registration Request to the MMS over the "selected" interface. In the Request line of this message there is the address of the correspondent interface of MMS.

When the 200 OK is received, the "keep-in-touch" mechanism is activated on that interface (and deactivated on the previous interface if needed). This procedure is activated at the start up of the MT (or when the MT first enters in a coverage area), or whenever the MT wants to change the selected interface if it is under coverage of more than one network. We can refer to this procedure as "off-call" mobility management because we assume that terminal is not engaged in a call. If the terminal is engaged in a call, the handover procedure will be executed (see later on).

Note that the use of different IP addresses in the MMS corresponding to the different interfaces of the Mobile Terminal is not needed from the point of view of the mobility

management procedure. The reason to use the two IP addresses is related to a routing problem in the Mobile Terminal. Using two IP addresses on the MMS is a convenient way in order to easily select on the Mobile Terminal which outgoing physical interface will be used for sending out the IP packets (both signaling and media packets).

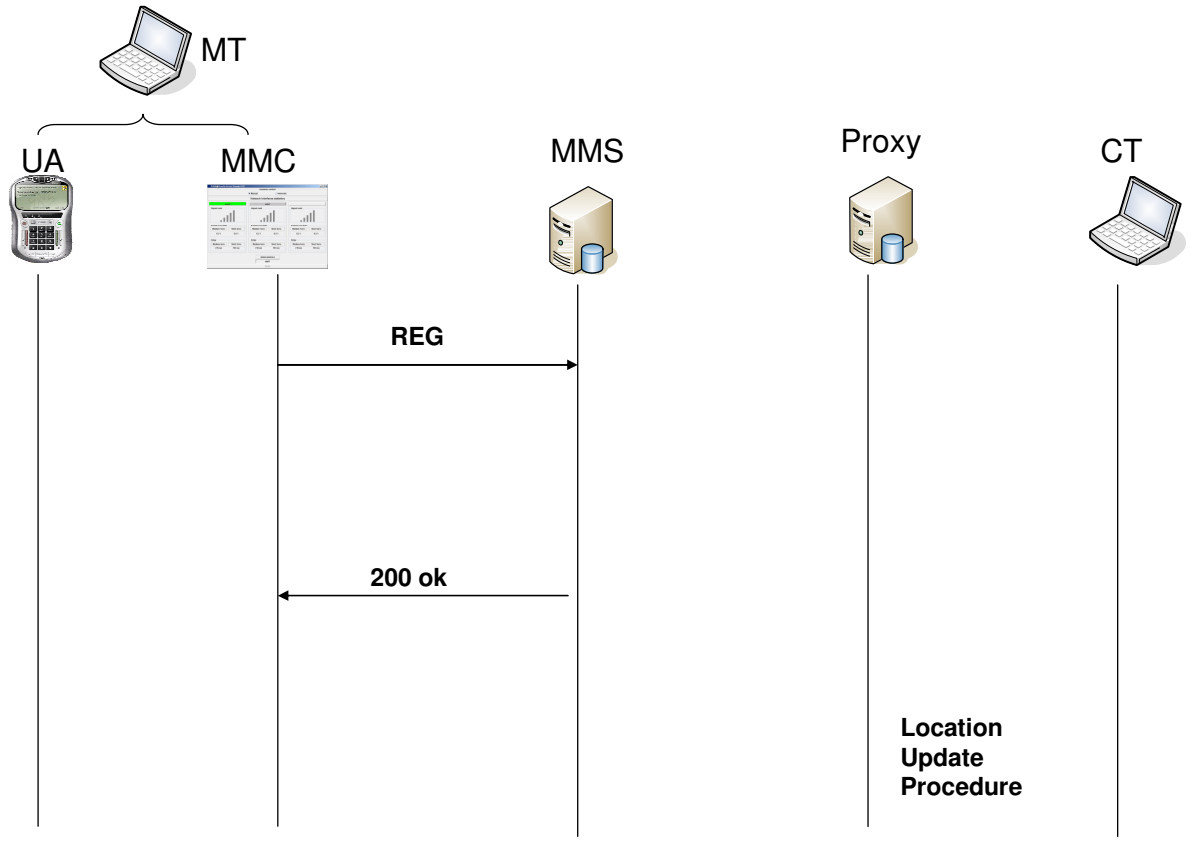


Figure 2.5: The Location Update Procedure

As result of the Location Update Registration procedure, the MMS becomes aware of the current position (i.e. IP address and port) of the MT, and can correctly route any new request or response messages addressed to the mobile UA (even across a NAT box). For each registered user the MMS stores the IP address and port from which it received the "Location Update" (LU) REGISTER. This information can be stored by the MMS in a table that we call "MMS mobility database" containing the MMS state. Such table is depicted in Table 2.2

User(terminal)	IP/port
user@domain.com	160.80.81.23:45678
user2@domain.com	87.3.235.212:23458

Table 2.2: MMS mobility database

In section 2.3.3 and 2.3.4 we have introduced the problem of UA identification and addressing. In order to identify the UA, we used an identifier that the MMC inserts in the Contact and in the Via header fields, and it is denoted as MMID (Mobility Management IDentifier) in the SIP messages shown in Table 2.3

Register MMC to MMS
REGISTER sip:160.80.82.27:5070 SIP/2.0 Via: SIP/2.0/UDP 83.225.138.116;MMID=user@iptel.uniroma2.it;branch=z9hG4bKd7bd1 Max-Forwards: 70 To: <sip: user@iptel.uniroma2.it > From: <sip: user@iptel.uniroma2.it >;tag=4758d7f7 Call-ID: 4614a25233b6f9f5@user CSeq: 1 REGISTER Contact: <sip:user@83.225.138.116> Expires: 3600 Content-Length: 0
200OK MMS to MMC
SIP/2.0 200 OK Via: SIP/2.0/UDP 83.225.138.116;MMID=user@iptel.uniroma2.it;received=83.225.138.116 To:<sip: user@iptel.uniroma2.it > From:<sip: user@iptel.uniroma2.it>;tag=4758d7f7 Call-ID: 4614a25233b6f9f5@user CSeq: 1 REGISTER Content-Length: 0

Table 2.3: Location Update SIP Request

2.4.1.1 "Keep-in-touch" mechanism for NAT traversal

A "keep-in-touch" mechanism is needed to keep the pinhole in the NAT open. Various techniques can be used such as dummy UDP packets (from the MMC to the MMS or vice-versa), mal-formed SIP messages, well-formed SIP messages. This is a typical function of Session Border Controllers. We use periodic Location Update messages from MMC to MMS. The "keep-in-touch" packets are sent every 30 seconds, so they use a very limited amount of resources.

2.4.2 User Registration

This procedure consists in the UA registration with its own SIP Registrar server (the backend SIP registrar). The sequence diagram of this procedure is described in Figure 2.6. As any other SIP message, when the UA sends its own registration request to the SIP Registrar,

the message is sent by the UA to the MMC which is seen as outbound proxy. In the Request line of this message the IP address (or its fully qualified domain name) of the SIP Registrar is indicated. The MMC forwards it to the MMS. Acting on behalf of the MT, the MMS will forward the registration to the SIP Registrar, which will update the contact address associated with the user's AoR (that is the public user identifier). When forwarding the Register message, the MMS/SBC modifies the Contact header in such a way it becomes the new "contact" for the user. This is required in order to force the routing through the SBC/MMS of all further requests addressed to the user. Such mangling of the contact URL should be unique and reversible. It can be done in several ways, using either a stateless approach (e.g. by mapping the previous URL, opportunely stuffed, within the new URL) or a stateful one (e.g. by using a local mapping table). We have chosen a stateless approach. The message "Registrar MMS to Proxy" of shows a rewritten contact. The contact has the format: <sip:/MMUSE-user/AT-83.225.138.116/PORT-5060@160.80.82.26:5070>. As defined in section 2.3.4.1 it includes a Token that can be chosen arbitrarily (we used MMUSE), then it includes the "Mobility Management ID" of the UA as defined above.

From now on, only the MMS will keep track of the MT movements, while the SIP Registrar will just believe that the MT location is the IP address of the SBC as indicated in the contact URL.

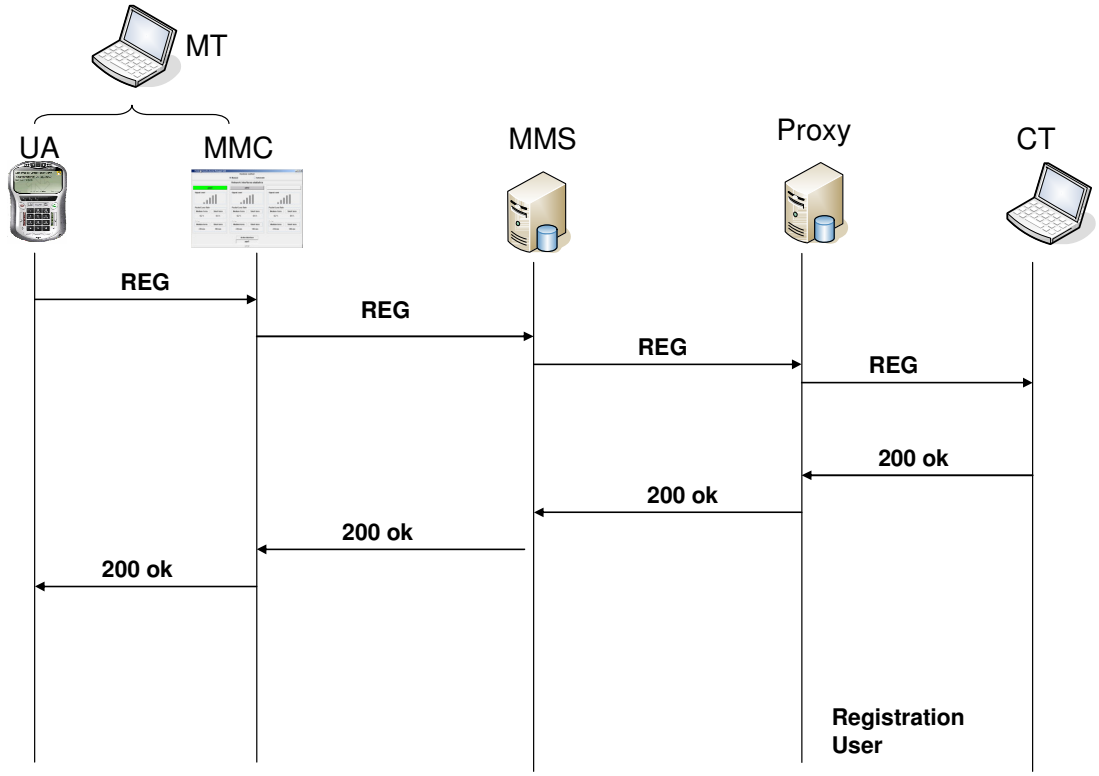


Figure 2.6: User Registration

REGISTER UA to MMC

```
REGISTER sip:iptel.uniroma2.it SIP/2.0
Via: SIP/2.0/UDP
83.225.138.116:5060;rport;branch=z9hG4bK39A9DAE6366344A494795BAC9B3D7C5C
From: user <sip:user@iptel.uniroma2.it>;tag=2872944525
To: user <sip:user@iptel.uniroma2.it>
Contact: "user" <sip:user@83.225.138.116:5060>
Call-ID: DEA9AAF40B7A454A9BA9FC037A12FEB9@iptel.uniroma2.it
CSeq: 1858 REGISTER
Expires: 1800
Max-Forwards: 70
User-Agent: X-Lite release 1105x
Content-Length: 0
```

REGISTER MMC to MMS

```
REGISTER sip:iptel.uniroma2.it SIP/2.0
Via: SIP/2.0/UDP 83.225.138.116;MMID=user@iptel.uniroma2.it;branch=z9hG4bKd7bd1
Via: SIP/2.0/UDP
83.225.138.116:5060;rport;branch=z9hG4bK39A9DAE6366344A494795BAC9B3D7C5C;receiv
ed=127.0.0.1
From: user <sip:user@iptel.uniroma2.it>;tag=2872944525
To: user <sip:user@iptel.uniroma2.it>
Call-ID: DEA9AAF40B7A454A9BA9FC037A12FEB9@iptel.uniroma2.it
CSeq: 1858 REGISTER
Expires: 1800
Max-Forwards: 69
User-Agent: X-Lite release 1105x
Contact: "user" <sip:user@83.225.138.116:5060>
Content-Length: 0
```

REGISTER MMS to Proxy

```
REGISTER sip:iptel.uniroma2.it:5061 SIP/2.0
Via: SIP/2.0/UDP 160.80.82.26:5070;branch=z9hG4bK443b8d64e
Via: SIP/2.0/UDP
83.225.138.116;MMID=user@iptel.uniroma2.it;branch=z9hG4bKd7bd1;received=83.225.
138.116
Route: <sip:iptel.uniroma2.it:5061;lr>
Via: SIP/2.0/UDP
83.225.138.116:5060;rport;branch=z9hG4bK39A9DAE6366344A494795BAC9B3D7C5C;receiv
ed=127.0.0.1
From: user <sip:user@iptel.uniroma2.it>;tag=2872944525
To: user <sip:user@iptel.uniroma2.it>
Call-ID: DEA9AAF40B7A454A9BA9FC037A12FEB9@iptel.uniroma2.it
CSeq: 1858 REGISTER
Expires: 1800
Max-Forwards: 68
User-Agent: X-Lite release 1105x
Contact: "user" <sip:/MMUSE-user/AT-83.225.138.116/PORT-5060@160.80.82.26:5070>
Content-Length: 0
```

200OK Proxy to MMS
SIP/2.0 200 OK Via: SIP/2.0/UDP 160.80.82.26:5070;branch=z9hG4bK443b8d64e Via: SIP/2.0/UDP 83.225.138.116;MMID=user@iptel.uniroma2.it;branch=z9hG4bKd7bd1;received=83.225. 138.116 Via: SIP/2.0/UDP 83.225.138.116:5060;rport;branch=z9hG4bK39A9DAE6366344A494795BAC9B3D7C5C;receiv ed=127.0.0.1 To: user <sip:user@iptel.uniroma2.it> From: user <sip:user@iptel.uniroma2.it>;tag=2872944525 Call-ID: DEA9AAF40B7A454A9BA9FC037A12FEB9@iptel.uniroma2.it CSeq: 1858 REGISTER Server: mjsip stack 1.6 Contact: "user" <sip:/MMUSE-user/AT-83.225.138.116/PORT- 5060@160.80.82.26:5070>;expires=1800 Content-Length: 0
200OK MMS to MMC
SIP/2.0 200 OK Via: SIP/2.0/UDP 83.225.138.116;MMID=user@iptel.uniroma2.it;branch=z9hG4bKd7bd1;received=83.225. 138.116 Via: SIP/2.0/UDP 83.225.138.116:5060;rport;branch=z9hG4bK39A9DAE6366344A494795BAC9B3D7C5C;receiv ed=127.0.0.1 To: user <sip:user@iptel.uniroma2.it> From: user <sip:user@iptel.uniroma2.it>;tag=2872944525 Call-ID: DEA9AAF40B7A454A9BA9FC037A12FEB9@iptel.uniroma2.it CSeq: 1858 REGISTER Server: mjsip stack 1.6 Contact: "user" <sip:user@83.225.138.116:5060>;expires=1800 Content-Length: 0
200OK MMC to UA
SIP/2.0 200 OK Via: SIP/2.0/UDP 83.225.138.116:5060;rport;branch=z9hG4bK39A9DAE6366344A494795BAC9B3D7C5C;receiv ed=127.0.0.1 To: user <sip:user@iptel.uniroma2.it> From: user <sip:user@iptel.uniroma2.it>;tag=2872944525 Call-ID: DEA9AAF40B7A454A9BA9FC037A12FEB9@iptel.uniroma2.it CSeq: 1858 REGISTER Server: mjsip stack 1.6 Contact: "user" <sip:user@83.225.138.116:5060>;expires=1800 Content-Length: 0

Table 2.4: User Registration Messages

2.4.3 Session Establishment

The session establishment procedure consists in a standard SIP session setup procedure. All session establishment messages for MT are handled by the MMS/SBC. Before relaying an INVITE request sent by the caller and the corresponding 200 OK response sent by the callee the SBC modifies the corresponding SDP bodies in order to act as RTP proxy for media flows in both directions. This is needed to correctly handle NAT traversal in the path towards the MT, and it is done by exploiting the symmetric RTP approach as in a typical SBC implementation.

Once the session is established, the media packets start to flow over the selected wireless interface. In principle, there is no need to send anything on the unselected active interfaces, that should be used only when an “on-call” mobility procedure occurs. The MMS needs to keep a state information related to the active flows as it is performing a media relay functionality. In order to correctly perform the handover procedure, we require that this state information is accessible using the current call as key. SIP identifies a call by means of the Call-ID, the From and To header fields. Therefore the MMS maintains an "MMS call database". For each call and for each media flow the information of two "legs" (MT-MMS and CT-MMS) needs to be stored. For each leg the local and remote IP addresses and port of the media flows are stored. On the other hand our practical experience suggested that starting sending the packets on the 3G interface introduces an initial delay that can be quite large and can cause noticeable disruption in the voice communication during the handoff. Therefore we introduce a “keep-alive” mechanism between MMC and MMS during the call phase: the MMC sends dummy UDP packets to the MMS over the unselected wireless interfaces. The MMS will take care of discarding these packets.

2.4.3.1 Outgoing Call

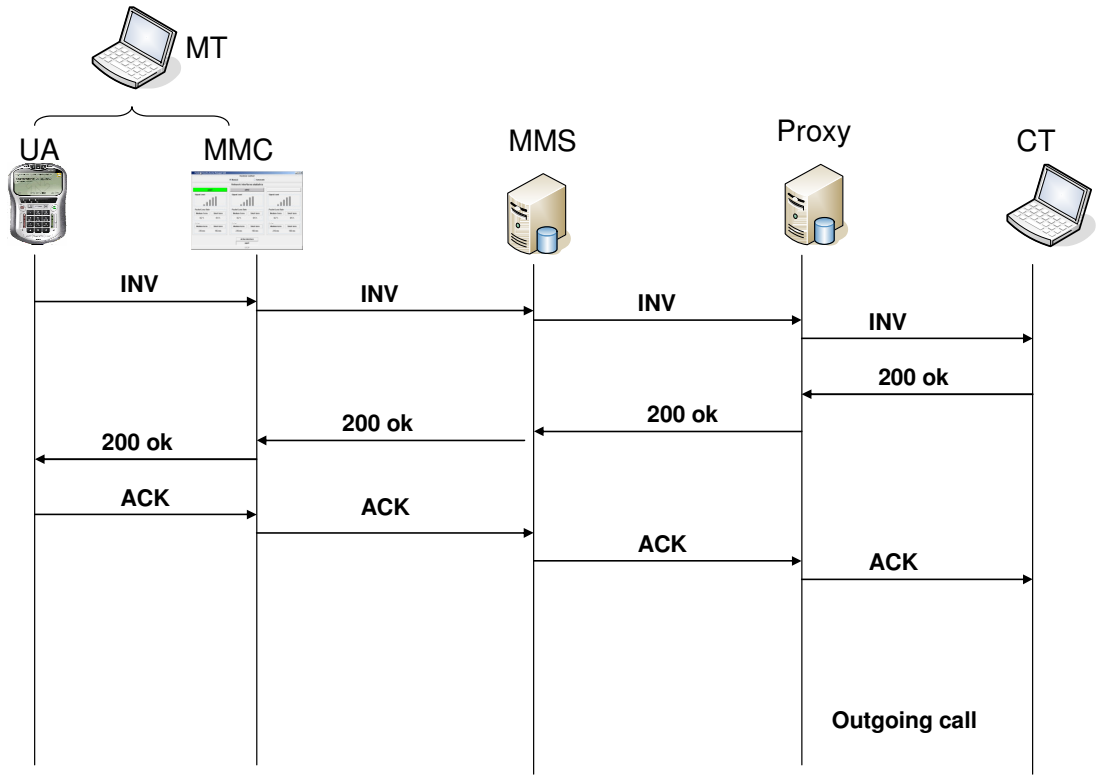


Figure 2.7: Outgoing Call

INVITE UA to MMC

```
INVITE sip:bob@iptel.uniroma2.it SIP/2.0
Via: SIP/2.0/UDP
83.225.138.116:5060;rport;branch=z9hG4bK1ADA35128CF14270B5126D698E0D74AD
From: user <sip:user@iptel.uniroma2.it>;tag=2723341418
To: <sip:bob@iptel.uniroma2.it>
Contact: <sip:user@83.225.138.116:5060>
Call-ID: BBED7847-4FA4-44B5-BE04-DB37FB664D2E@83.225.138.116
CSeq: 50930 INVITE
Max-Forwards: 70
Content-Type: application/sdp
User-Agent: X-Lite release 1105x
Content-Length: 210

v=0
o=user 687448 687499 IN IP4 83.225.138.116
s=X-Lite
c=IN IP4 83.225.138.116
t=0 0
m=audio 8000 RTP/AVP 3 101
a=rtpmap:3 gsm/8000
a=rtpmap:101 telephone-event/8000
a=fmtp:101 0-15
a=sendrecv
```

INVITE MMC to MMS

```
INVITE sip: bob@iptel.uniroma2.it SIP/2.0
Via: SIP/2.0/UDP 83.225.138.116;MMID=user@iptel.uniroma2.it;branch=z9hG4bKd7bd1
Via: SIP/2.0/UDP
83.225.138.116:5060;rport;branch=z9hG4bK1ADA35128CF14270B5126D698E0D74AD;receiv
ed=127.0.0.1
From: user <sip:user@iptel.uniroma2.it>;tag=2723341418
To: <sip:bob@iptel.uniroma2.it>
Call-ID: BBED7847-4FA4-44B5-BE04-DB37FB664D2E@83.225.138.116
CSeq: 50930 INVITE
Max-Forwards: 69
Contact: <sip:user@83.225.138.116:5060>
User-Agent: X-Lite release 1105x
Content-Length: 211
Content-Type: application/sdp

v=0
o=user 687448 687499 IN IP4 83.225.138.116
s=X-Lite
c=IN IP4 83.225.138.116
t=0 0
m=audio 10000 RTP/AVP 3 101
a=rtpmap:3 gsm/8000
a=rtpmap:101 telephone-event/8000
a=fmtp:101 0-15
a=sendrecv
```

INVITE MMS to Proxy

```
INVITE sip: bob@iptel.uniroma2.it SIP/2.0
Via: SIP/2.0/UDP 160.80.82.26:5070;branch=z9hG4bK2f7b09664
Via: SIP/2.0/UDP
83.225.138.116;MMID=user@iptel.uniroma2.it;branch=z9hG4bKd7bd1;received=83.225.
138.116
Route: <sip:iptel.uniroma2.it:5061;lr>
Via: SIP/2.0/UDP
83.225.138.116:5060;rport;branch=z9hG4bK1ADA35128CF14270B5126D698E0D74AD;receiv
ed=127.0.0.1
From: user <sip:user@iptel.uniroma2.it>;tag=2723341418
To: <sip:bob@iptel.uniroma2.it>
Call-ID: BBED7847-4FA4-44B5-BE04-DB37FB664D2E@83.225.138.116
CSeq: 50930 INVITE
Record-Route: <sip:160.80.82.26:5070;lr>
Max-Forwards: 68
User-Agent: X-Lite release 1105x
Contact: <sip:/MMUSE-user/AT-83.225.138.116/PORT-5060@160.80.82.26:5070>
Content-Length: 209
Content-Type: application/sdp

v=0
o=user 687448 687499 IN IP4 83.225.138.116
s=X-Lite
c=IN IP4 160.80.82.27
t=0 0
m=audio 10074 RTP/AVP 3 101
```

```
a=rtpmap:3 gsm/8000
a=rtpmap:101 telephone-event/8000
a=fmtp:101 0-15
a=sendrecv
```

INVITE Proxy to CT

```
INVITE sip:bob@160.80.82.81:5060 SIP/2.0
Via: SIP/2.0/UDP
iptel.uniroma2.it:5061;rport;branch=z9hG4bKd0e0d6cc168e581d94c910e4
Via: SIP/2.0/UDP 160.80.82.26:5070;branch=z9hG4bK2f7b09664
Via: SIP/2.0/UDP
83.225.138.116;MMID=user@iptel.uniroma2.it;branch=z9hG4bKd7bd1;received=83.225.
138.116
Via: SIP/2.0/UDP
83.225.138.116:5060;rport;branch=z9hG4bK1ADA35128CF14270B5126D698E0D74AD;receiv
ed=127.0.0.1
From: user <sip:user@iptel.uniroma2.it>;tag=2723341418
To: <sip:bob@iptel.uniroma2.it>
Call-ID: BBED7847-4FA4-44B5-BE04-DB37FB664D2E@83.225.138.116
CSeq: 50930 INVITE
Record-Route: <sip:iptel.uniroma2.it:5061;lr>
Record-Route: <sip:160.80.82.26:5070;lr>
Max-Forwards: 67
User-Agent: X-Lite release 1105x
Contact: <sip:/MMUSE-user/AT-83.225.138.116/PORT-5060@160.80.82.26:5070>
Content-Length: 209
Content-Type: application/sdp

v=0
o=user 687448 687499 IN IP4 83.225.138.116
s=X-Lite
c=IN IP4 160.80.82.27
t=0 0
m=audio 10074 RTP/AVP 3 101
a=rtpmap:3 gsm/8000
a=rtpmap:101 telephone-event/8000
a=fmtp:101 0-15
```

200 OK CT to Proxy

```
SIP/2.0 200 Ok
Via: SIP/2.0/UDP
iptel.uniroma2.it:5061;rport;branch=z9hG4bKd0e0d6cc168e581d94c910e4
Via: SIP/2.0/UDP 160.80.82.26:5070;branch=z9hG4bK2f7b09664
Via: SIP/2.0/UDP
83.225.138.116;MMID=user@iptel.uniroma2.it;branch=z9hG4bKd7bd1;received=83.225.
138.116
Via: SIP/2.0/UDP
83.225.138.116:5060;rport;branch=z9hG4bK1ADA35128CF14270B5126D698E0D74AD;receiv
ed=127.0.0.1
From: user <sip:user@iptel.uniroma2.it>;tag=2723341418
To: <sip:bob@iptel.uniroma2.it>;tag=923999554
Contact: <sip:bob@160.80.82.81:5060>
Record-Route: <sip:iptel.uniroma2.it:5061;lr>,<sip:160.80.82.26:5070;lr>
Call-ID: BBED7847-4FA4-44B5-BE04-DB37FB664D2E@83.225.138.116
CSeq: 50930 INVITE
Content-Type: application/sdp
Server: X-Lite release 1103m
```

```
Content-Length: 242

v=0
o=bob 571000 579750 IN IP4 160.80.82.81
s=X-Lite
c=IN IP4 160.80.82.81
t=0 0
m=audio 8000 RTP/AVP 3 0 8 101
a=rtpmap:0 pcmu/8000
a=rtpmap:8 pcma/8000
a=rtpmap:3 gsm/8000
a=rtpmap:101 telephone-event/8000
a=fmtp:101 0-15
```

200 OK Proxy to MMS

```
SIP/2.0 200 Ok
Via: SIP/2.0/UDP 160.80.82.26:5070;branch=z9hG4bK2f7b09664
Via: SIP/2.0/UDP
83.225.138.116;MMID=user@iptel.uniroma2.it;branch=z9hG4bKd7bd1;received=83.225.
138.116
Via: SIP/2.0/UDP
83.225.138.116:5060;rport;branch=z9hG4bK1ADA35128CF14270B5126D698E0D74AD;receiv
ed=127.0.0.1
From: user <sip:user@iptel.uniroma2.it>;tag=2723341418
To: <sip:bob@iptel.uniroma2.it>;tag=923999554
Contact: <sip:bob@160.80.82.81:5060>
Record-Route: <sip:iptel.uniroma2.it:5061;lr>,<sip:160.80.82.26:5070;lr>
Call-ID: BBED7847-4FA4-44B5-BE04-DB37FB664D2E@83.225.138.116
CSeq: 50930 INVITE
Content-Type: application/sdp
Server: X-Lite release 1103m
Content-Length: 242

v=0
o=bob 571000 579750 IN IP4 160.80.82.81
s=X-Lite
c=IN IP4 160.80.82.81
t=0 0
m=audio 8000 RTP/AVP 3 0 8 101
a=rtpmap:0 pcmu/8000
a=rtpmap:8 pcma/8000
a=rtpmap:3 gsm/8000
a=rtpmap:101 telephone-event/8000
a=fmtp:101 0-15
```

200 OK MMS to MMC

```
SIP/2.0 200 Ok
Via: SIP/2.0/UDP
83.225.138.116;MMID=user@iptel.uniroma2.it;branch=z9hG4bKd7bd1;received=83.225.
138.116
Via: SIP/2.0/UDP
83.225.138.116:5060;rport;branch=z9hG4bK1ADA35128CF14270B5126D698E0D74AD;receiv
ed=127.0.0.1
From: user <sip:user@iptel.uniroma2.it>;tag=2723341418
To: <sip:bob@iptel.uniroma2.it>;tag=923999554
Record-Route: <sip:iptel.uniroma2.it:5061;lr>,<sip:160.80.82.26:5070;lr>
Call-ID: BBED7847-4FA4-44B5-BE04-DB37FB664D2E@83.225.138.116
```

```
CSeq: 50930 INVITE
Server: X-Lite release 1103m
Contact: <sip:/MMUSE-user/AT-83.225.138.116/PORT-5060@160.80.82.26:5070>
Content-Length: 243
Content-Type: application/sdp

v=0
o=bob 571000 579750 IN IP4 160.80.82.81
s=X-Lite
c=IN IP4 160.80.82.27
t=0 0
m=audio 10076 RTP/AVP 3 0 8 101
a=rtpmap:0 pcmu/8000
a=rtpmap:8 pcma/8000
a=rtpmap:3 gsm/8000
a=rtpmap:101 telephone-event/8000
a=fmtp:101 0-15
```

200 OK MMC to UA

```
SIP/2.0 200 Ok
Via: SIP/2.0/UDP
83.225.138.116:5060;rport;branch=z9hG4bK1ADA35128CF14270B5126D698E0D74AD;receiv
ed=127.0.0.1
From: user <sip:user@iptel.uniroma2.it>;tag=2723341418
To: <sip:bob@iptel.uniroma2.it>;tag=923999554
Call-ID: BBED7847-4FA4-44B5-BE04-DB37FB664D2E@83.225.138.116
CSeq: 50930 INVITE
Record-Route: <sip:iptel.uniroma2.it:5061;lr>
Record-Route: <sip:127.0.0.1:5050;lr>
Server: X-Lite release 1103m
Contact: <sip:/MMUSE-user/AT-83.225.138.116/PORT-5060@160.80.82.26:5070>
Content-Length: 245
Content-Type: application/sdp

v=0
o=bob 571000 579750 IN IP4 160.80.82.81
s=X-Lite
c=IN IP4 83.225.138.116
t=0 0
m=audio 10002 RTP/AVP 3 0 8 101
a=rtpmap:0 pcmu/8000
a=rtpmap:8 pcma/8000
a=rtpmap:3 gsm/8000
a=rtpmap:101 telephone-event/8000
a=fmtp:101 0-15
```

ACK UA to MMC

```
ACK sip:bob@iptel.uniroma2.it SIP/2.0
Via: SIP/2.0/UDP
83.225.138.116:5060;rport;branch=z9hG4bK4827DB6F62A147B685699175FD699690
From: user <sip:user@iptel.uniroma2.it>;tag=2723341418
To: <sip:bob@iptel.uniroma2.it>;tag=923999554
Contact: <sip:user@83.225.138.116:5060>
Route: <sip:127.0.0.1:5050;lr>,<sip:iptel.uniroma2.it:5061;lr>
Call-ID: BBED7847-4FA4-44B5-BE04-DB37FB664D2E@83.225.138.116
CSeq: 50930 ACK
Max-Forwards: 70
```

Content-Length: 0
ACK MMC to MMS
ACK sip:bob@iptel.uniroma2.it SIP/2.0 Via: SIP/2.0/UDP 83.225.138.116;MMID=user@iptel.uniroma2.it;branch=z9hG4bKd7bd1 Via: SIP/2.0/UDP 83.225.138.116:5060;rport;branch=z9hG4bK4827DB6F62A147B685699175FD699690;receiv ed=127.0.0.1 Route: <sip:iptel.uniroma2.it:5061;lr> From: user <sip:user@iptel.uniroma2.it>;tag=2723341418 To: <sip:bob@iptel.uniroma2.it>;tag=923999554 Call-ID: BBED7847-4FA4-44B5-BE04-DB37FB664D2E@83.225.138.116 CSeq: 50930 ACK Max-Forwards: 69 Contact: <sip:user@83.225.138.116:5060> Content-Length: 0
ACK MMS to Proxy
ACK sip:bob@iptel.uniroma2.it SIP/2.0 Via: SIP/2.0/UDP 160.80.82.26:5070;branch=z9hG4bK9ca495347 Via: SIP/2.0/UDP 83.225.138.116;MMID=user@iptel.uniroma2.it;branch=z9hG4bKd7bd1;received=83.225. 138.116 Via: SIP/2.0/UDP 83.225.138.116:5060;rport;branch=z9hG4bK4827DB6F62A147B685699175FD699690;receiv ed=127.0.0.1 Route: <sip:iptel.uniroma2.it:5061;lr> From: user <sip:user@iptel.uniroma2.it>;tag=2723341418 To: <sip:bob@iptel.uniroma2.it>;tag=923999554 Call-ID: BBED7847-4FA4-44B5-BE04-DB37FB664D2E@83.225.138.116 CSeq: 50930 ACK Max-Forwards: 68 Contact: <sip:/MMUSE-user/AT-83.225.138.116/PORT-5060@160.80.82.26:5070> Content-Length: 0
ACK Proxy to CT
ACK sip:bob@160.80.82.81:5060 SIP/2.0 Via: SIP/2.0/UDP iptel.uniroma2.it:5061;rport;branch=z9hG4bKd9e13bcc029a1995450abe51 Via: SIP/2.0/UDP 160.80.82.26:5070;branch=z9hG4bK9ca495347 Via: SIP/2.0/UDP 83.225.138.116;MMID=user@iptel.uniroma2.it;branch=z9hG4bKd7bd1;received=83.225. 138.116 Via: SIP/2.0/UDP 83.225.138.116:5060;rport;branch=z9hG4bK4827DB6F62A147B685699175FD699690;receiv ed=127.0.0.1 From: user <sip:user@iptel.uniroma2.it>;tag=2723341418 To: <sip:bob@iptel.uniroma2.it>;tag=923999554 Call-ID: BBED7847-4FA4-44B5-BE04-DB37FB664D2E@83.225.138.116 CSeq: 50930 ACK Max-Forwards: 67 Contact: <sip:/MMUSE-user/AT-83.225.138.116/PORT-5060@160.80.82.26:5070> Content-Length: 0

Table 2.5: Outgoing call SIP Messages

2.4.3.2 Incoming Call

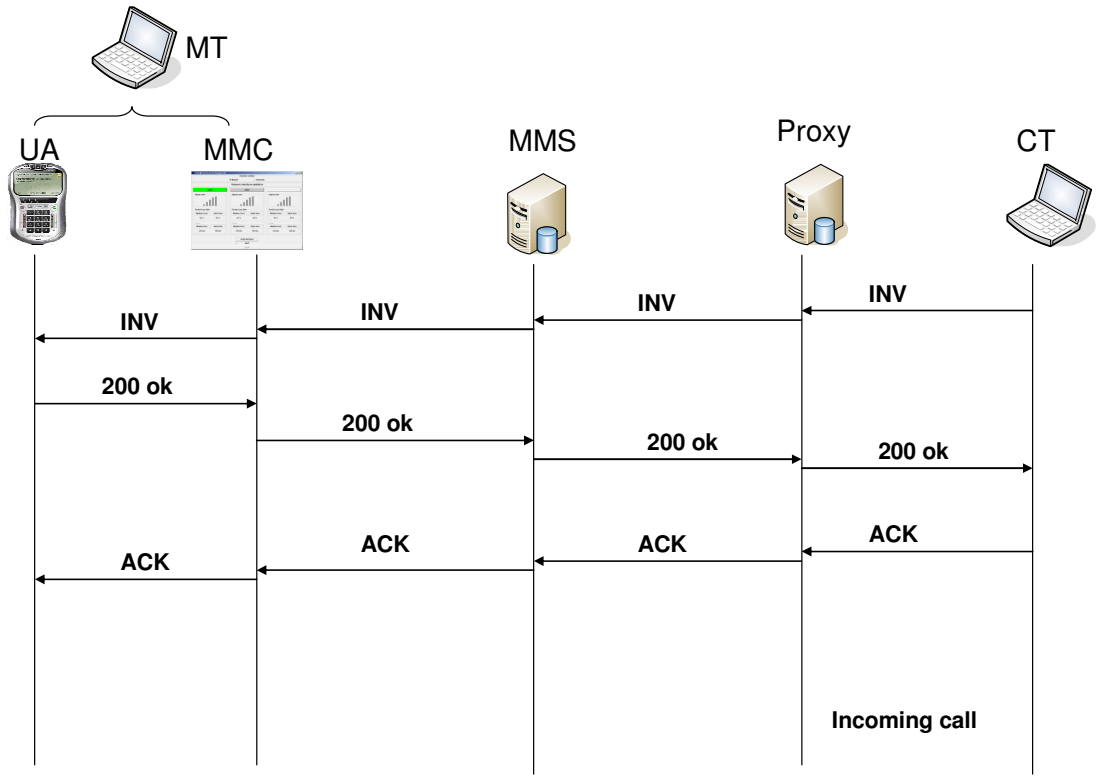


Figure 2.8: Incoming Call

INVITE CT to Proxy

```
INVITE sip:user@iptel.uniroma2.it SIP/2.0
Via: SIP/2.0/UDP
160.80.82.81:5060;rport;branch=z9hG4bKB3CD8472A8754926A525B00C4D150CF9
From: bob <sip:bob@iptel.uniroma2.it>;tag=834828371
To: <sip:user@iptel.uniroma2.it>
Contact: <sip:bob@160.80.82.81:5060>
Call-ID: AF045434-4AED-4248-8B4B-10E0474CDCA9@160.80.82.81
CSeq: 10936 INVITE
Max-Forwards: 70
Content-Type: application/sdp
User-Agent: X-Lite release 1103m
Content-Length: 242

v=0
o=bob 632875 632875 IN IP4 160.80.82.81
s=X-Lite
c=IN IP4 160.80.82.81
t=0 0
m=audio 8000 RTP/AVP 0 8 3 101
a=rtpmap:0 pcmu/8000
a=rtpmap:8 pcma/8000
a=rtpmap:3 gsm/8000
a=rtpmap:101 telephone-event/8000
a=fmtp:101 0-15
```

INVITE Proxy to MMS

```
INVITE sip:/MMUSE-user/AT-83.225.138.116/PORT-5060@160.80.82.26:5070 SIP/2.0
Via: SIP/2.0/UDP
iptel.uniroma2.it:5061;rport;branch=z9hG4bKfbe62746c92fc293be0df388
Via: SIP/2.0/UDP
160.80.82.81:5060;branch=z9hG4bKB3CD8472A8754926A525B00C4D150CF9;rport=5060
From: bob <sip:bob@iptel.uniroma2.it>;tag=834828371
To: <sip:user@iptel.uniroma2.it>
Contact: <sip:bob@160.80.82.81:5060>
Call-ID: AF045434-4AED-4248-8B4B-10E0474CDCA9@160.80.82.81
CSeq: 10936 INVITE
Record-Route: <sip:iptel.uniroma2.it:5061;lr>
Max-Forwards: 69
Content-Type: application/sdp
User-Agent: X-Lite release 1103m
Content-Length: 242

v=0
o=bob 632875 632875 IN IP4 160.80.82.81
s=X-Lite
c=IN IP4 160.80.82.81
t=0 0
m=audio 8000 RTP/AVP 0 8 3 101
a=rtpmap:0 pcmu/8000
a=rtpmap:8 pcma/8000
a=rtpmap:3 gsm/8000
a=rtpmap:101 telephone-event/8000
a=fmtp:101 0-15
```

INVITE MMS to MMC

```
INVITE sip:user@83.225.138.116:5060 SIP/2.0
Via: SIP/2.0/UDP 160.80.82.26:5070;branch=z9hG4bKd5e708e7f
Via: SIP/2.0/UDP
iptel.uniroma2.it:5061;rport;branch=z9hG4bKfbe62746c92fc293be0df388
Via: SIP/2.0/UDP
160.80.82.81:5060;branch=z9hG4bKB3CD8472A8754926A525B00C4D150CF9;rport=5060
From: bob <sip:bob@iptel.uniroma2.it>;tag=834828371
To: <sip:user@iptel.uniroma2.it>
Call-ID: AF045434-4AED-4248-8B4B-10E0474CDCA9@160.80.82.81
CSeq: 10936 INVITE
Record-Route: <sip:160.80.82.26:5070;lr>
Record-Route: <sip:iptel.uniroma2.it:5061;lr>
Max-Forwards: 68
User-Agent: X-Lite release 1103m
Contact: < sip:/MMUSE-user/AT-83.225.138.116/PORT-5060@160.80.82.26:5070>
Content-Length: 243
Content-Type: application/sdp

v=0
o=bob 632875 632875 IN IP4 160.80.82.81
s=X-Lite
c=IN IP4 160.80.82.27
t=0 0
m=audio 10078 RTP/AVP 0 8 3 101
a=rtpmap:0 pcmu/8000
a=rtpmap:8 pcma/8000
```

```
a=rtpmap:3 gsm/8000
a=rtpmap:101 telephone-event/8000
a=fmtp:101 0-15
```

INVITE MMC to UA

```
INVITE sip:user@83.225.138.116:5060 SIP/2.0
Via: SIP/2.0/UDP 127.0.0.1:5050;branch=z9hG4bK2b1c69da2
Via: SIP/2.0/UDP
160.80.82.26:5070;branch=z9hG4bKd5e708e7f;received=160.80.82.27
Via: SIP/2.0/UDP
iptel.uniroma2.it:5061;rport;branch=z9hG4bKfbe62746c92fc293be0df388
Via: SIP/2.0/UDP
160.80.82.81:5060;branch=z9hG4bKB3CD8472A8754926A525B00C4D150CF9;rport=5060
From: bob <sip:bob@iptel.uniroma2.it>;tag=834828371
To: <sip:user@iptel.uniroma2.it>
Call-ID: AF045434-4AED-4248-8B4B-10E0474CDCA9@160.80.82.81
CSeq: 10936 INVITE
Max-Forwards: 67
User-Agent: X-Lite release 1103m
Contact: < sip:/MMUSE-user/AT-83.225.138.116/PORT-5060@160.80.82.26:5070>
Content-Length: 245
Content-Type: application/sdp

v=0
o=bob 632875 632875 IN IP4 160.80.82.81
s=X-Lite
c=IN IP4 83.225.138.116
t=0 0
m=audio 10004 RTP/AVP 0 8 3 101
a=rtpmap:0 pcmu/8000
a=rtpmap:8 pcma/8000
a=rtpmap:3 gsm/8000
a=rtpmap:101 telephone-event/8000
a=fmtp:101 0-15
```

200OK UA to MMC

```
SIP/2.0 200 Ok
Via: SIP/2.0/UDP 127.0.0.1:5050;branch=z9hG4bK2b1c69da2;received=127.0.0.1
Via: SIP/2.0/UDP
160.80.82.26:5070;branch=z9hG4bKd5e708e7f;received=160.80.82.27
Via: SIP/2.0/UDP
iptel.uniroma2.it:5061;rport;branch=z9hG4bKfbe62746c92fc293be0df388
Via: SIP/2.0/UDP
160.80.82.81:5060;branch=z9hG4bKB3CD8472A8754926A525B00C4D150CF9;rport=5060
From: bob <sip:bob@iptel.uniroma2.it>;tag=834828371
To: <sip:user@iptel.uniroma2.it>;tag=1724648780
Contact: <sip:user@83.225.138.116:5060>
Call-ID: AF045434-4AED-4248-8B4B-10E0474CDCA9@160.80.82.81
CSeq: 10936 INVITE
Content-Type: application/sdp
Server: X-Lite release 1105x
Content-Length: 210

v=0
o=user 751083 758842 IN IP4 83.225.138.116
s=X-Lite
c=IN IP4 83.225.138.116
```

```
t=0 0
m=audio 8000 RTP/AVP 3 101
a=rtpmap:3 gsm/8000
a=rtpmap:101 telephone-event/8000
a=fmtp:101 0-15
a=sendrecv
```

200OK MMC to MMS

```
SIP/2.0 200 Ok
Via: SIP/2.0/UDP
160.80.82.26:5070;branch=z9hG4bKd5e708e7f;received=160.80.82.27
Via: SIP/2.0/UDP
iptel.uniroma2.it:5061;rport;branch=z9hG4bKfbe62746c92fc293be0df388
Via: SIP/2.0/UDP
160.80.82.81:5060;branch=z9hG4bKB3CD8472A8754926A525B00C4D150CF9;rport=5060
From: bob <sip:bob@iptel.uniroma2.it>;tag=834828371
To: <sip:user@iptel.uniroma2.it>;tag=1724648780
Call-ID: AF045434-4AED-4248-8B4B-10E0474CDCA9@160.80.82.81
CSeq: 10936 INVITE
Contact: <sip:user@83.225.138.116:5060>
Server: X-Lite release 1105x
Content-Length: 211
Content-Type: application/sdp

v=0
o=user 751083 758842 IN IP4 83.225.138.116
s=X-Lite
c=IN IP4 83.225.138.116
t=0 0
m=audio 10006 RTP/AVP 3 101
a=rtpmap:3 gsm/8000
a=rtpmap:101 telephone-event/8000
a=fmtp:101 0-15
a=sendrecv
```

200OK MMS to Proxy

```
SIP/2.0 200 Ok
Via: SIP/2.0/UDP
iptel.uniroma2.it:5061;rport;branch=z9hG4bKfbe62746c92fc293be0df388
Via: SIP/2.0/UDP
160.80.82.81:5060;branch=z9hG4bKB3CD8472A8754926A525B00C4D150CF9;rport=5060
From: bob <sip:bob@iptel.uniroma2.it>;tag=834828371
To: <sip:user@iptel.uniroma2.it>;tag=1724648780
Call-ID: AF045434-4AED-4248-8B4B-10E0474CDCA9@160.80.82.81
CSeq: 10936 INVITE
Server: X-Lite release 1105x
Contact: <sip:/MMUSE-user/AT-83.225.138.116/PORT-5060@160.80.82.26:5070>
Content-Length: 209
Content-Type: application/sdp

v=0
o=user 751083 758842 IN IP4 83.225.138.116
s=X-Lite
c=IN IP4 160.80.82.27
t=0 0
m=audio 10080 RTP/AVP 3 101
a=rtpmap:3 gsm/8000
```

```
a=rtpmap:101 telephone-event/8000
a=fmtp:101 0-15
a=sendrecv
```

200OK Proxy to CT

```
SIP/2.0 200 Ok
Via: SIP/2.0/UDP
160.80.82.81:5060;branch=z9hG4bKB3CD8472A8754926A525B00C4D150CF9;rport=5060
From: bob <sip:bob@iptel.uniroma2.it>;tag=834828371
To: <sip:user@iptel.uniroma2.it>;tag=1724648780
Call-ID: AF045434-4AED-4248-8B4B-10E0474CDCA9@160.80.82.81
CSeq: 10936 INVITE
Server: X-Lite release 1105x
Contact: <sip:/MMUSE-user/AT-83.225.138.116/PORT-5060@160.80.82.26:5070>
Content-Length: 209
Content-Type: application/sdp

v=0
o=user 751083 758842 IN IP4 83.225.138.116
s=X-Lite
c=IN IP4 160.80.82.27
t=0 0
m=audio 10080 RTP/AVP 3 101
a=rtpmap:3 gsm/8000
a=rtpmap:101 telephone-event/8000
a=fmtp:101 0-15
a=sendrecv
```

ACK CT to Proxy

```
ACK sip:user@iptel.uniroma2.it SIP/2.0
Via: SIP/2.0/UDP
160.80.82.81:5060;rport;branch=z9hG4bK73383B1E52AE418E82B9855AA104BA62
From: bob <sip:bob@iptel.uniroma2.it>;tag=834828371
To: <sip:user@iptel.uniroma2.it>;tag=1724648780
Contact: <sip:bob@160.80.82.81:5060>
Call-ID: AF045434-4AED-4248-8B4B-10E0474CDCA9@160.80.82.81
CSeq: 10936 ACK
Max-Forwards: 70
Content-Length: 0
```

ACK Proxy to MMS

```
ACK sip:/MMUSE-user/AT-83.225.138.116/PORT-5060@160.80.82.26:5070 SIP/2.0
Via: SIP/2.0/UDP
iptel.uniroma2.it:5061;rport;branch=z9hG4bKfb9900fefd36c0e43f37ac2e
Via: SIP/2.0/UDP
160.80.82.81:5060;branch=z9hG4bK73383B1E52AE418E82B9855AA104BA62;rport=5060
From: bob <sip:bob@iptel.uniroma2.it>;tag=834828371
To: <sip:user@iptel.uniroma2.it>;tag=1724648780
Contact: <sip:bob@160.80.82.81:5060>
Call-ID: AF045434-4AED-4248-8B4B-10E0474CDCA9@160.80.82.81
CSeq: 10936 ACK
Max-Forwards: 69
Content-Length: 0
```

ACK MMS to MMC

```
ACK sip: bob@83.225.138.116:5060 SIP/2.0
Via: SIP/2.0/UDP 160.80.82.26:5070;branch=z9hG4bKef9148c07
```

Via: SIP/2.0/UDP iptel.uniroma2.it:5061;rport;branch=z9hG4bKfb9900fefd36c0e43f37ac2e Via: SIP/2.0/UDP 160.80.82.81:5060;branch=z9hG4bK73383B1E52AE418E82B9855AA104BA62;rport=5060 From: bob <sip:bob@iptel.uniroma2.it>;tag=834828371 To: <sip:user@iptel.uniroma2.it>;tag=1724648780 Call-ID: AF045434-4AED-4248-8B4B-10E0474CDCA9@160.80.82.81 CSeq: 10936 ACK Max-Forwards: 68 Contact: < sip:/MMUSE-user/AT-83.225.138.116/PORT-5060@160.80.82.26:5070> Content-Length: 0
ACK MMC to UA
ACK sip:user@83.225.138.116:5060 SIP/2.0 Via: SIP/2.0/UDP 127.0.0.1:5050;branch=z9hG4bK352433cbe Via: SIP/2.0/UDP 160.80.82.26:5070;branch=z9hG4bKef9148c07;received=160.80.82.27 Via: SIP/2.0/UDP iptel.uniroma2.it:5061;rport;branch=z9hG4bKfb9900fefd36c0e43f37ac2e Via: SIP/2.0/UDP 160.80.82.81:5060;branch=z9hG4bK73383B1E52AE418E82B9855AA104BA62;rport=5060 From: bob <sip:bob@iptel.uniroma2.it>;tag=834828371 To: <sip:user@iptel.uniroma2.it>;tag=1724648780 Call-ID: AF045434-4AED-4248-8B4B-10E0474CDCA9@160.80.82.81 CSeq: 10936 ACK Max-Forwards: 67 Contact: < sip:/MMUSE-user/AT-83.225.138.116/PORT-5060@160.80.82.26:5070> Content-Length: 0

Table 2.6: Incoming Call SIP messages

2.4.4 On-Call Mobility: the Handover procedure

The on-call mobility management procedure takes place when the UA identifies the need for handoff during an ongoing session. In our proposal, all the handover signaling messages can be exchanged on the target network (this approach is commonly referred to as "forward" handover). Therefore the handover can be performed even if the communication on the old network is interrupted abruptly. The handover procedure (see Figure 2.9) is MT initiated. The MMC in the terminal sends an “handover” Register message over the target network interface addressed to the MMS in the SBC. The MMC in the MT sends an "HandOver" (HO) REGISTER over the target network interface addressed to the MMS. Differently from a "Location Update" (LU) REGISTER, the "HandOver" (HO) REGISTER request contains in the message header the reference to the active session(s) to which the handover is referred.

At the same time, the MT starts duplicating the outgoing media packets on both interfaces (unless the old interface has gone down). As soon as the MMS receives the "HandOver" (HO) REGISTER, it starts accepting packets coming from the new interface and discarding the ones coming from the old interface for the active session(s) to be handed over. Then it sends back

the 200 OK to the MMC and it starts sending the media packets directed to the MT using the new interface. Thanks to the fact that the terminal has already started sending the packets on the new interface, the duration of the handover is minimized.

The most critical issue is that the "HandOver" (HO) REGISTER could be lost for any reason, delaying the handoff procedure. The standard SIP procedure foresees that the client performs a set of retransmission of the "HandOver" (HO) REGISTER if the 200 OK is not received back Since the retransmission timeout and the retransmission duration suggests in the rfc 3261 (T1=500 ms and 64*T1 respectively) are not compatible with a reasonable performance of the handover in case of the loss of the "HandOver" (HO) REGISTER, we use the lower timers (T1 = 50 ms) for the "HandOver" (HO) REGISTER. On the MT side, the MMC will stop duplicating the packets on both interfaces as soon as the 200 OK is received or the first media packet is received on the new interface.

Note that if the media packet is received, but no 200 OK message, the MMC will still continue sending the Register message until the Register transaction expires.

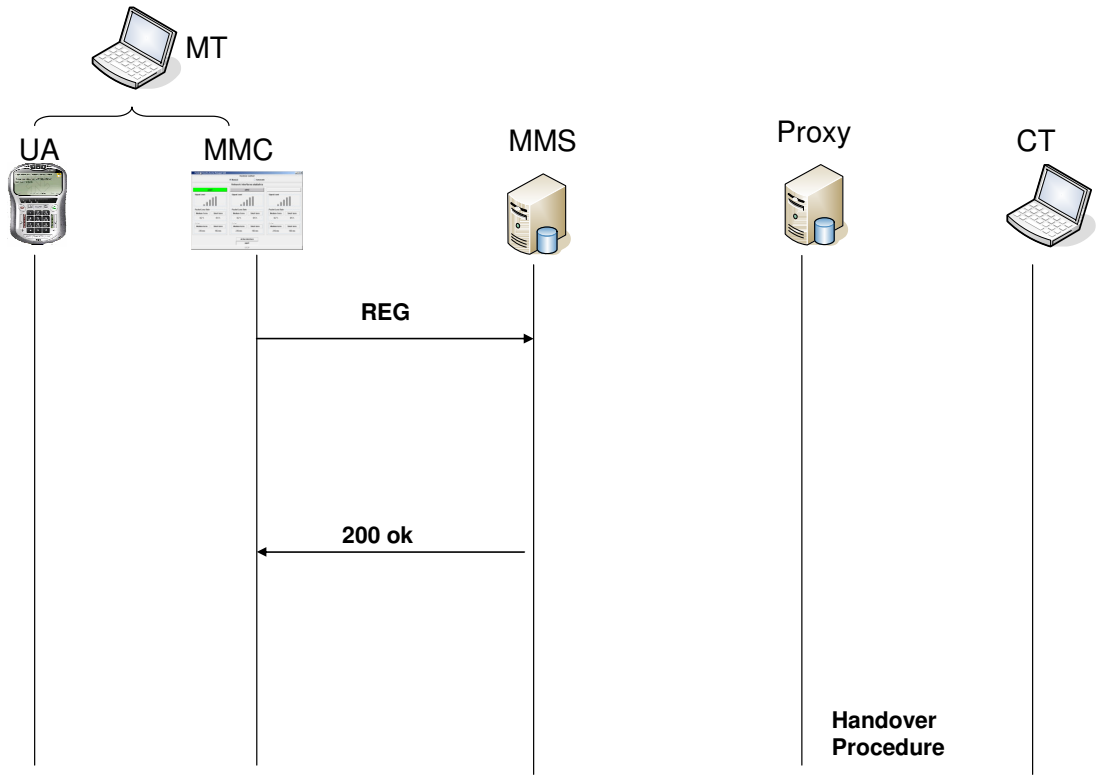


Figure 2.9: Handover Procedure

As previously described, in order to perform an handover during an active call, we need to include in the HO Register information about the call to be handed over. This information are contained in a new header. The new header name will be "Handover:" and it will carry the

Call-ID and the two tags, in a similar way to the Join header [rfc3911] or to the Replaces header [rfc3891]. In particular the parameters that carry the tags are called "req-tag" and "other-tag".

Handover: sxdfv20000513@host.domain.com; req-tag=erfg; other-tag=wdfe

Table 2.7 shows an (HO) Registrar example.

Register MMC to MMS
REGISTER sip:160.80.82.26:5070 SIP/2.0 Via: SIP/2.0/UDP 83.225.138.116;MMID=user@iptel.uniroma2.it;branch=z9hG4bKd7bd1 Max-Forwards: 70 To: <sip:user@iptel.uniroma2.it> From: <sip:user@iptel.uniroma2.it>;tag=erfg **Handover: AF0454344AED42488B4B10E04A9@160.80.82.81; req-tag=erfg; other-tag=wdfe** Call-ID: 4614a25233b6f9f5@user CSeq: 2 REGISTER Contact: <sip:user@83.225.138.116> Expires: 3600 Content-Length: 0
200OK MMS to MMC
SIP/2.0 200 OK Via: SIP/2.0/UDP 83.225.138.116;MMID=user@iptel.uniroma2.it;received=83.225.138.116 To: <sip:user@iptel.uniroma2.it> From: <sip:user@iptel.uniroma2.it>;tag=erfg Call-ID: 4614a25233b6f9f5@user CSeq: 1 REGISTER Content-Length: 0

Table 2.7: SIP messages for handover procedure

The MMC requesting the handover will insert in the "req-tag" the tag corresponding to the MT in the INVITE transaction that originated the call (i.e. the From: tag if the call was originated by the MT or the To: tag if the call was originated by the CT). By comparing this information with the MMS call database, the MMS will be able to understand on which of the two legs of the call the handover has been requested.

2.5 Handover Criteria

In this section we describe how the MMC can take the decision to perform a handover. Several criteria can be used (also in combination) to drive the handover decision as for example the signal quality at the radio level or the cost of sending/receiving traffic over the access networks or power saving.

In our implementation we consider 3 kinds of handover criteria:

- Quality of signal
- Quality of link
- Location

All these criteria are not exclusive, it is possible to combine them in order to improve the handover decision. It is important to consider that proper hysteresis mechanism should be included to prevent frequent switching from one access network to another.

2.5.1 Quality of Signal

The quality of signal approach consists of a periodical monitoring of the quality of the received signal (power, S/N ratio) from each wireless network interface (typically this information is contained in the network card drivers). The advantage of this solution is that the client can perform this information without any “help” from the network just reading its drivers.

The problem is that signal quality information is only local and does not take into account the load of the wireless cell nor the load of the network between the cell and the SBC. For this reason we introduced a mechanism to make IP level measurements of the QoS (Quality of Services) in the path between the MT and the SBC over the different wireless networks as shown in next section.

2.5.2 Quality of Link

In this Section we present efficient mechanisms to evaluate loss and round trip time in order to measure the QoS in the network. The QoS parameters considered in this work are the RTT (Round Trip Time), the jitter and the packet loss.

In order to take the needed measurements both a passive and an active measurement approach could be used. Where possible, passive measurements are better as they introduce

less overhead. Our solution is a mix of passive and active measurements. We try to evaluate loss using a passive approach relying on the RTP packets that carry the VoIP flows, whenever these RTP packets are sent. When RTP packets are not sent we need to send additional probe packets (i.e. on the active wireless channel due to silence suppression, or if we want to measure loss on an inactive wireless channel which does not carry the VoIP flow). The procedure also foresee to piggyback loss information evaluated on the server side on the RTP packets that are sent from the MMS to the MMC (if the RTP packets are being sent, otherwise they are inserted in the additional probe packets. As for the RTT estimation, the procedure that will be detailed in this section foresees to send timestamp information from MMC to MMS and vice versa. Similarly, this information can be piggybacked on the RTP packets if they are present or inserted in the additional probe packets. Therefore we have the following types of packet: *RTPplain*, *RTPprobe*, *SAprobe* (Stand Alone probe).

2.5.2.1 Generic definitions for loss and RTT evaluation

The loss estimation needs to be done between MMC and MMS in both directions, so we will denote with superscript “up” the “upstream” direction from the MMC to the MMS and with superscript “down” the “downstream” direction from the MMS to the MMC.

Figure 2.10 shows the messages exchanged between MMC and MMS according to the following notation:

- $Ts(seq^{up})$ is the time when MMC sends the MMCprobe packet
- $Tr(Rseq^{up})$ is the time when the MMS receives the MMCprobe
- $Ts(Rseq^{down})$ is the time when the MMS sends the MMSprobe packet
- $Tr(Rseq^{down})$ is the time when the MMC receives the MMSprobe packet

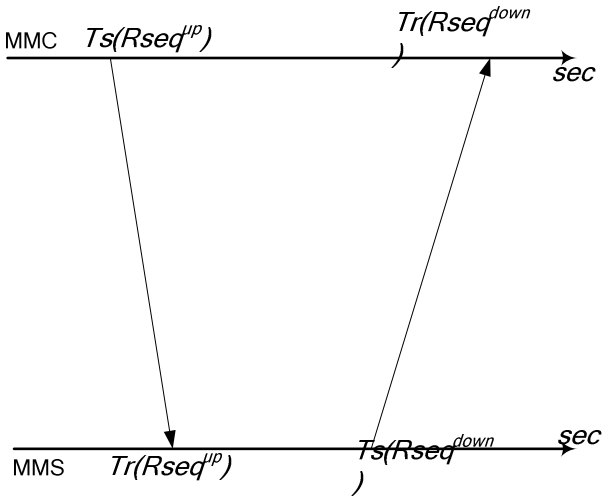


Figure 2.10: Probes packet

The MMC keeps a single counter called *currentSeq*^{up} which is increased for each packet (*RTPplain*, *RTPprobe* or *SAprobe*) sent towards the MMS over a given UDP connection. The MMC sets a timer of period T ms, when the timer expires the timer is set again with duration T ms and more over another timer of period αT (α<1) ms is set. If there is an RTP packet being sent from MMC to MMS before the αT timer expires, the RTP packet is extended and it becomes an *RTPprobe* packet. If the αT timer expires, a *SAprobe* packet is sent. Similarly the MMS keeps a counter called *currentSeq*^{down} and it uses the same procedure for sending *RTPprobe* or *SAprobe* packets.

The *RTPprobe* as well as the *SAprobe* packets sent by the MMC contains the sequence number of uplink packets (*Seq*^{up}) and the timestamp when the packet is sent denoted by *Ts(Seq*^{up}*)*. $MMCprobe[Seq^{up};Ts(seq^{up})]$

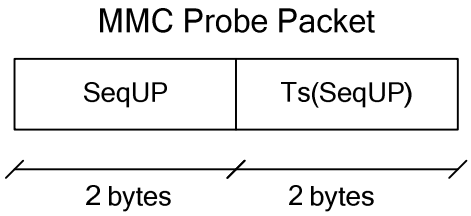


Figure 2.11: The MMC probe packet

While sending an *RTPprobe* or a *SAprobe* packet, the MMC stores the following tuple in a list which is ordered on the *Seq*^{up} value and denoted *Uplink-probes-list*:

Seq^{up}	$Ts(Seq^{up})$

Figure 2.12: Uplink-probes-list

For any packet received from the MMC, the MMS increases a counter denoted $TotRec^{up}$, which accumulates the total number of received packets. Then, if the packet is an MMCprobe packet (either RTPprobe or SAprobe), it stores the following information in a state variable *Last-MMCprobe-received:*

$Rseq^{up}$	$\mathrm{Ts}(Rseq^{up})$	$\mathrm{Tr}(Rseq^{up})$	$\mathrm{TotRec}(\mathrm{Tr}(Rseq^{up}))$

Figure 2.13 Last-MMCprobe-received

where Rseq^{up} and $\mathrm{Ts}(\mathrm{Rseq}^{up})$ are respectively the Seq^{up} and the $\mathrm{Ts}(\mathrm{Seq}^{up})$ contained in the received packet, $\mathrm{Tr}(\mathrm{Rseq}^{up})$ is the timestamp when the packet has been received and $\mathrm{TotRec}^{up}(\mathrm{Tr}(\mathrm{Rseq}^{up}))$ is the counter of received packets at the time the packet is received.

When the MMS sends an MMS Probe packet (RTP or SA), it includes the sequence number for downlink packets Seq^{down}, the timestamp when the packet is $(\mathrm{Ts}(\mathrm{Seq}^{down}))$, the value of the counter TotRec^{up} at the time the packet is sent and the state information stored upon the receival of the last MMC probe packet.

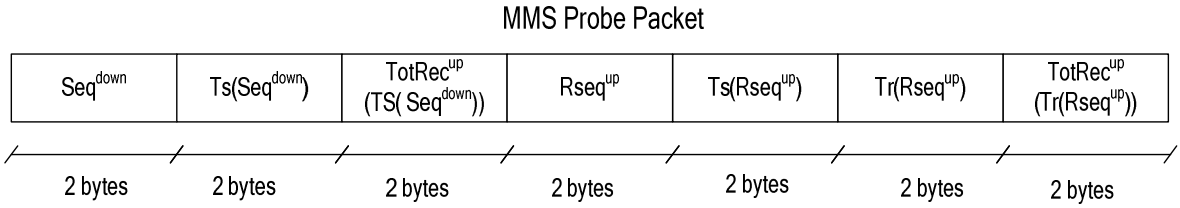


Figure 2.14: The MMS probe packet

When the MMC receives any packet from the MMS, it increases a counter denoted TotRec^{down}, which accumulates the total number of received packets. Then, if the packet is an *MMSprobe* packet (either *RTPprobe* or *SAprobe*), it stores the following tuple in a list which is ordered on the Seq^{down} value and denoted *Downlink-probes-list*:

$Rseq^{down}$	$\mathrm{Tr}(Rseq^{down})$	$\mathrm{TotRec}^{down}(\mathrm{Tr}(Rseq^{down}))$

Figure 2.15: The Downlink-probes-list

2.5.2.2 Uplink Loss Estimation

When the MMC receives an RTPprobe or an SAprobe, the Rseq^{up} contained in the packets is matched with the Seq^{up} value in the *Uplink-probes-list*. The tuple in the list is extended adding $\mathrm{Tr}(\mathrm{Rseq}^{up})$ i.e. the time (as perceived by the MMS) when the uplink probe packet Seq^{up} was received by the MMS and $\mathrm{TotRec}^{up}(\mathrm{Tr}(\mathrm{Rseq}^{up}))$, i.e. the accumulated number of packets received by the MMS at that time.

Moreover the following information is stored (it is a single tuple that is overwritten for each received packet and that is denoted *Downlink-recent-probe*):

Seq^{down}	$\mathrm{Ts}(Seq^{down})$	$\mathrm{TotRec}(\mathrm{T}s(Seq^{down}))$

Figure 2.16: The Downlink-recent-probe

The uplink loss is estimated on the MMC side as follows. Assume that $t_{client} = t_{now}$ and that we want to estimate the loss on a window of duration not larger than $t_{winloss}$.

We browse the *Uplink-probes-list* until we find a Seq^{up} such that $Ts(Seq^{up}) > t_{now} - t_{winloss}$, we denote it Seq^{up}_{start}. Then we browse the *Uplink-probes-list* up to the most recent packet that has been acknowledged by an MMS probe (i.e. the tuple in the list includes $TotRec^{up}(Tr(Rseq^{up}))$, we denote it Seq^{up}_{acked}).

The evaluation of the loss perceived by the MMS in the interval [$Tr(Rseq^{up}_{start})$, $Tr(Rseq^{up}_{acked})$] is estimated by the MMC as:

$$uplinkLoss = \min\left(1 - \frac{\mathrm{Received}^{up}}{\mathrm{Sent}^{up}}, 0\right)$$

where the number of sent packets in upstream ($Sent^{up}$) is evaluated as:

$$Sent^{up} = Seq^{up}_{acked} - Seq^{up}_{start}$$

$Sent^{up} = Seq^{up}_{acked} - Seq^{up}_{start}$,

while the number of received packets in upstream ($Received^{up}$) as:

$$Received^{up} = \mathrm{TotRec}^{up}(\mathrm{Tr}(\mathrm{Rseq}^{up}_{acked})) - \mathrm{TotRec}^{up}(\mathrm{Tr}(\mathrm{Rseq}^{up}_{start}))$$

This procedure works smoothly if the MMC probes are not lost in their path towards the MMS. If the MMS does not receive a set of successive MMC probes, the information related to the most recent acknowledged packet becomes “old”. Note that in this case the MMS keeps sending the MMS probe packets repeating the $Rseq^{up}$; $Ts(Rseq^{up})$; $Tr(Rseq^{up})$; $TotRec^{up}(Tr(Rseq^{up}))$ information which refers to the last received MMS probe packet, but it updates the $TotRec^{up}(Ts(seq^{down}))$ which refers to the accumulated number of received packets when the MMSprobe packet is sent. In this case, the loss evaluation can be corrected as follows, where $Ts(seq^{down})$; $TotRec^{up}(Ts(seq^{down}))$ are taken from *Downlink-recent-probe*:

$$Received^{up} = Received^{up} + \mathrm{TotRec}^{up}(\mathrm{Ts}(\mathrm{Seq}^{down})) - \mathrm{TotRec}^{up}(\mathrm{Tr}(\mathrm{Rseq}^{up}_{acked}))$$

$$Sent^{up} = Sent^{up} + \mathrm{FSeq}^{up}(\mathrm{Ts}(Seq^{up}_{acked}) + \mathrm{Ts}(Seq^{down}) - \mathrm{Tr}(Rseq^{up}_{acked})) - Seq^{up}_{acked}$$

where $FSeq^{up}(T)$ is a function that should provide the number of uplink packets that are sent at time T, where T is a continuous value. Unfortunately we only store this information in

Uplink-probes-list at the time instants when the MMC probes are sent. Therefore we can in general provide only a lower bound and an upper bound to this value. The lower bound $LBSeq^{up}$ is the greatest Seq^{up} in *Uplink-probes-list* such that $Ts(Seq^{up})<T$. The upper bound is $UBSeq^{up}$ is the smaller Seq^{up} in *Uplink-probes-list* such that $Ts(Seq^{up})>T$, or $currentSeq^{up}$ if there is not such Seq^{up}. Note that, due to jitter phenomena in the uplink channel, the $LBSeq^{up}$ and $UBSeq^{up}$ are not 100% guaranteed to be respectively a lower and upper bound of the packet sent in the time interval that we are considering for the loss evaluation. Anyway they are reasonably good for our purposes of keeping track in real time of the packet loss. While a linear approximation between the lower and upper bound is feasible, we have chosen to simply use the lower bound, which implies that a lower bound (in a probabilistic sense) on the loss is evaluated.

If *MMSprobe* packets are not arriving to the MMC the uplink loss estimates cannot be updated (and likely we are in a situation where the downlink loss is too high).

2.5.2.3 Downlink Loss Estimation

As told before the MMC stores information about the Probe packets received from the MMS: $Rseq^{down}$,$Tr(Rseq^{down})$ and $TotRec^{down}(Tr(Rseq^{down}))$ in the"Downlink-probes-list" (Figure 2.15).The downlink loss is estimated on the MMC side as follows.

Assume that $t_{client} = t_{now}$ and that we want to estimate the loss on a window of duration not larger than $t_{winloss}$. We browse the *Downlink-probes-list* until we find a $RSeq^{down}$ such that $Ts(Seq^{up}) > t_{now} - t_{winloss}$, we denote it $RSeq^{down}{}_{start}$. Then we browse the *Downlink-probes-list* up to the most recent received packet, we denote it $RSeq^{down}{}_{last}$.

The evaluation of the loss perceived by the MMC in the interval [$Tr(Rseq^{down}{}_{start})$, $Tr(Rseq^{down}{}_{last})$] is made by the MMC evaluating the number of $Sent^{down}$ packets as:

$Sent^{down} = RSeq^{down}{}_{last} - RSeq^{down}{}_{start}$,

and the number of received packets as:

$Received^{down} = TotRec^{down}(Tr(Rseq^{down}{}_{last})) - TotRec^{down}(Tr(Rseq^{down}{}_{start}))$,

$$uplinkLoss = \min\left(1 - \frac{\text{Received}^{\text{down}}}{\text{Sent}^{\text{down}}}, 0\right)$$

estimated by the MMC as:

where the number of sent packets in upstream (Sent^{up}) is evaluated as:

$$Sent^{up} = Seq^{up}_{acked} - Seq^{up}_{start}$$

Sent^{up} = Seq$^{up}_{acked}$- Seq$^{up}_{start}$,

while the number of received packets in upstream (Received^{up}) as:

$$Received^{up} = \mathrm{TotRec}^{up}(\mathrm{Tr}(\mathrm{Rseq}^{up}_{acked})) - \mathrm{TotRec}^{up}(\mathrm{Tr}(\mathrm{Rseq}^{up}_{start}))$$

We defined two temporal windows in which the loss ratio (uplink and downlink) is evaluated: a “short window” in the order of few seconds () and a “Medium window” in the order of few tens of seconds. The temporal window duration is set in the parameter $t_{winloss}$

2.5.2.4 RTT Estimation

To perform the RTT computation the MMC sends MMCprobe packets containing the upstream sequence number seq^{up} and time stamp Ts(seq^{up}). The MMS copies both the sequence number of the received packet Rseq^{up} and the previously defined time stamps Ts(Rseq^{up}), Tr(Rseq^{up}), Ts(Rseq^{down}) in the MMSprobe message (see Figure 2.14).

When MMC received a probe packet from MMS it compares the Seq^{up} of the last MMCprobe sent to MMS, and the Rseq^{up} in order to understand if RTT can be estimated. If the difference between is less than two packets, it starts to estimate the RTT.

First it calculates the current RTT (RTT_{ist}) as:

$$RTT_{ist} = \left(Tr(Seq_{down}) - Ts\left(Rseq_{up}\right)\right) - \left(Ts(Rseq_{down}) - Tr\left(Rseq_{up}\right)\right)$$

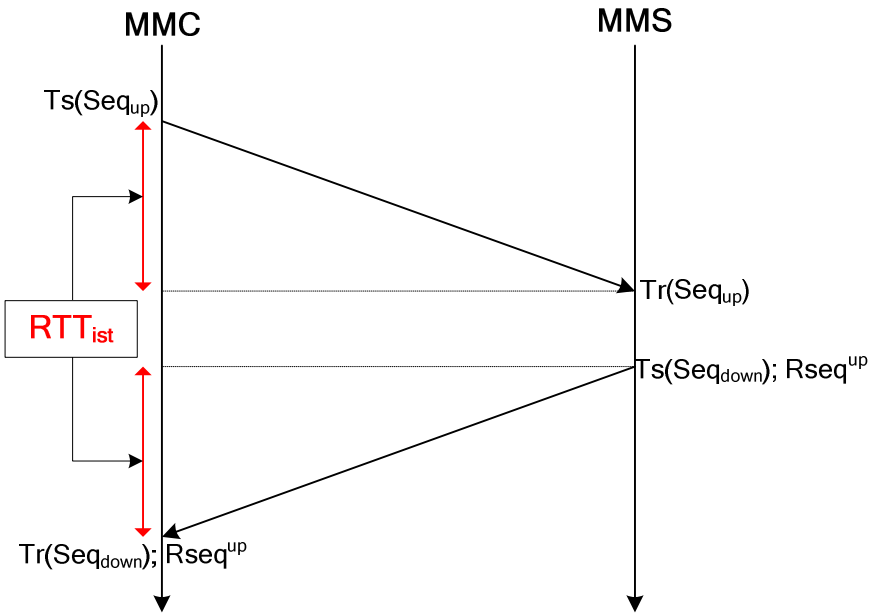


Figure 2.17: RTT estimation

Then it compares RTT_{ist} with the value of the RTT_{avg} previously calculated. The RTT_{avg} is defined as the average RTT in a temporal window.

If RTT_{ist} is more than RTT_{avg} than the RTT_{avg} became RTT_{ist} otherwise the RTT_{avg} is:

$$RTT_{avg} = ewma(RTT_{avg}, RTT_{ist}, (Rseq_{up} - Lseq_{up}), wt)$$

where:

- $Lseq_{up}$ is the sequence number of the last RTT_{ist} previuolsy calculated
- wt is the temporal window where RTT_{avg} is calculated (as well as the packet loss estimation we defined a medium and a short temporal windows)
- ewma is a Exponential Weighted Moving Average estimator defined as:

$$ewma(MA, New, w, S) = MA \cdot \frac{S - w}{S} + \frac{New \cdot w}{S}$$

2.5.2.5 Uplink Jitter Estimation

In order to estimate the Jitter during the probe packets (Figure 2.11 Figure 2.14) exchange the MMC needs to keep as state variables:

- $last\_ackd^{up}$: sequence number $Rseq^{up}$ contained in the last received MMSprobe packet
- $Tr^{up}(last\_ackd^{up})$, $Ts^{up}(last\_ackd^{up})$: time stamps for sending and receiving packet $Rseq^{up}$ contained in the last received MMSprobe packet
- $Jacc^{up}$: the "accumulated" jitter (i.e. the Jitter with respect to the first recived packets),
- $Jmin_{ST}^{up}$ and $Jmax_{ST}^{up}$: the minimum and maximum of $Jacc^{up}$ in the time window

The algorithm for Jitter estimation for a temporal window is the following:

```
If (Rsequp - last_ackdup) <= 0
   //out of sequence or deuplicated
Else
  If (Rsequp - last_ackdup) >= SjST
    // first packet after a long loss period
    Jaccup=0
    JminSTup = +∞
    JmaxSTup = -∞
  Else
    // here (Rsequp - last_ackdup) > 0
    // i.e. not duplicated nor out-of-sequence
    Jtempup=Trup(Rsequp)-
    + Trup_last_ackdupTsup(Rsequp)+Tsup_last_ackdup
    Jaccup= Jaccup+ Jtempup
    If Jaccup <= JminSTup
      JminSTup = Jaccup
```

```
    Else
      JminSTup=ewma(JminSTup,Jaccup,Rsequp-last_ackdup,SjST)
    Endif
    If Jaccup >= JmaxSTup
      JmaxSTup = Jaccup
    Else
      JmaxSTup=ewma(JmaxSTup,Jaccup,Rsequp-last_ackdup,SjST)
    Endif
    last_ackdup = Rsequp
  Endif
Endif
```

Where ewma is defined as:

$$ewma(MA, New, w, S) = MA \cdot \frac{S-w}{S} + \frac{New \cdot w}{S}$$

The estimations of the Jitter for the upstream direction will be:

$J_{ST}^{up} = Jmax_{ST}^{up} - Jmin_{ST}^{up}$
$J_{MT}^{up} = Jmax_{MT}^{up} - Jmin_{MT}^{up}$

Also for the estimation of Jitter we define a Short temporal window (ST) and a Medium Temporal window.

For the downlink the algorithm are the same just replacing “up” with “down”

2.5.3 Location-assisted

The location information could help the MMC to reduce the power consumption of the Mobile Terminal before and after the handover procedure. Suppose that the Mobile Terminal is a Smart Phone equipment with two network interfaces (3G and a WiFi) and a GPS receiver. Typically wifi coverage is limited to "hot spots" while 3G coverage is virtually ubiquitous, but Wifi connection is often faster and cheaper than 3G. So the user could like use WiFi access network each time is possible. This could imply to have 2 antennas always on in the smart phone. If we know a priori the WiFi hot spot position, and if we can know, thanks to the GPS receiver, position and speed of the phone, we can choose to turn on the WiFi interface just in “hot spot” proximity and only if the estimated “dwelling” time of the phone inside the hot spot area is significant (*crossing time*).

In this scenario we have to consider two case studies:

1. **Moving-out scenario**: The user is moving out of the WiFi hot spot area. At the edge of a WiFi area, the handoff algorithm based on the knowledge of the mobile terminal position, decide when to trigger the handoff procedure and *switch seamlessly* to an overlaying cellular data connection (UMTS).

2. **Moving-in scenario**: The user is moving in a WiFi area. Like for the first case study, the use of a GPS receiver allows the calculation of handoff metrics, in particular distance from the AP and time estimates. The handoff procedure is triggered on the basis of these parameters and permits to *switch seamlessly* to a WiFi link.

We use two handoff metrics to demonstrate the feasibility of the proposed solution:

- for "Moving-in scenario" the *distance* of the Mobile Terminal from the centre of the WiFi cell is constantly monitored; the "Mobile Manager" uses the algorithm shown in Figure 2.19 to trigger vertical handover.
- for "Moving-out scenario" the value of *crossing time* continuously estimated trigger vertical handover on the basis of the conditions shown in Figure 2.19.

First of all we introduce the modelling of the WiFi-cell and the procedure aiming to calculate the handoff metrics. WiFi coverage area has been modelled by a single cell with a circular area having a radius R (in meters). It is possible to evaluate:

- the distance of the Mobile Terminal from the centre of the circular area (we suppose that the Access Point is in the centre and its coverage area is circular);
- the direction of the Mobile Terminal on the basis of two consecutive positions;
- the boundary direction of the Mobile Terminal. This parameter allows to estimate if the user is approaching to the WiFi cell or not on the basis of the calculus of the angular cone. See the following figure for a graphical description of the model;
- two important parameters: the approaching time to the WiFi cell and the crossing time of the circular area on the basis of speed information of the Mobile Terminal.

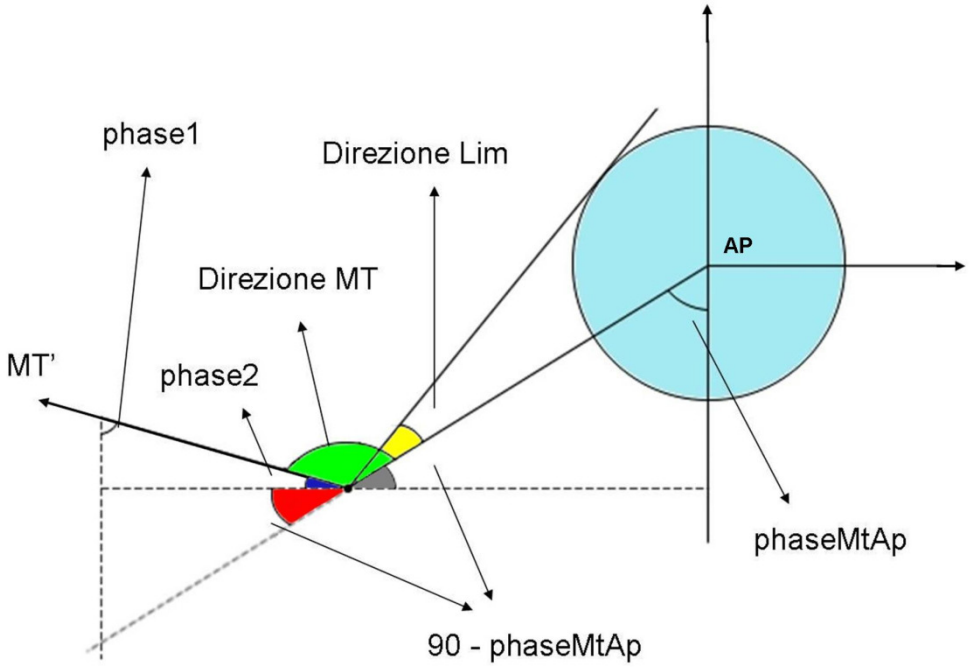


Figure 2.18: Scheme for Client direction evaluation

The following figure illustrates the vertical handoff procedure valid for our demonstration scenarios.

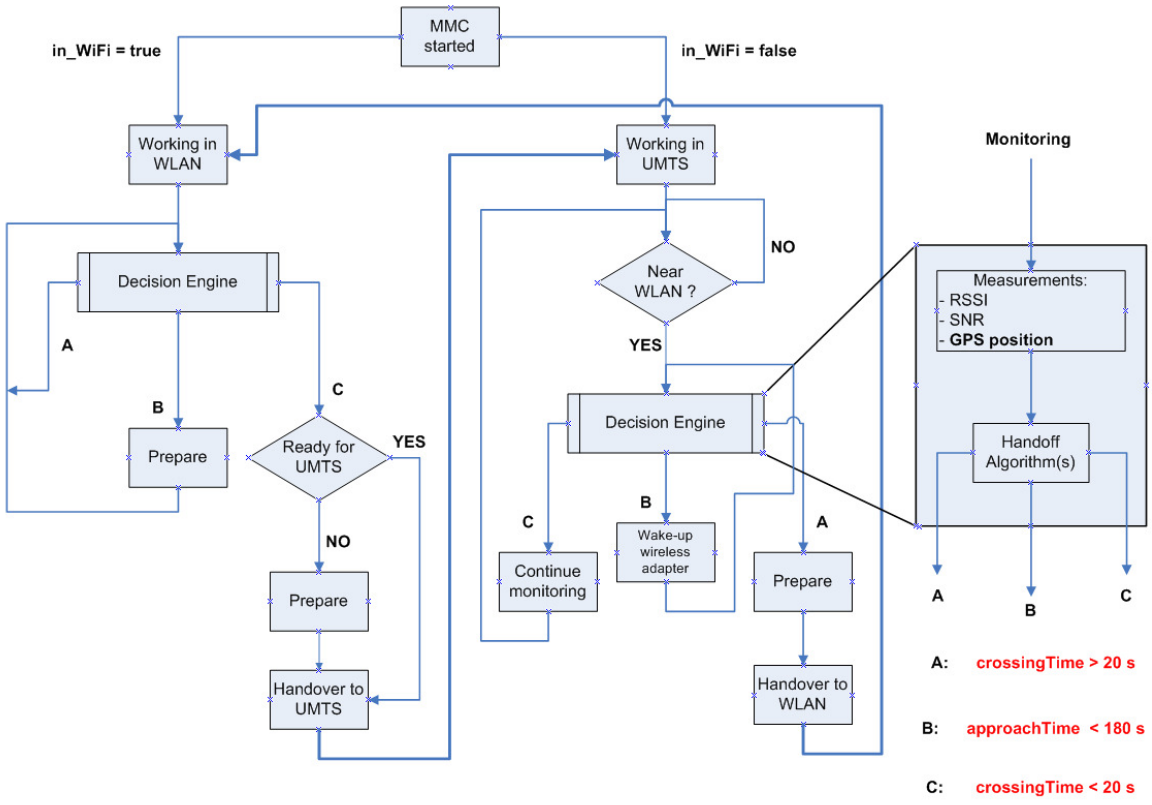


Figure 2.19: Localization Handover Algorithm

2.6 MMUSE Implementations and testbed

Two independent implementations exist both for the MMS and for the MMC. A first implementation of the MMS has been realized in Java jointly by the University of Roma "Tor Vergata" and the University of Parma, both in Italy, by using and modifying the open source MjSip Java SIP stack [33]. A second implementation has been made by NEC Network Laboratory in Heidelberg Germany, based on the open source SIP Express Router (SER) [34], developed in C, and whose functionalities have been extended to perform the tasks of the MMS. As for the MMC, there is a first implementation in Java realized by the two Universities quoted above, by using the same MjSip Java SIP stack, and a second implementation, made by BULL Italia (now Eutelia), written in C++ for WindowsMobile5.0, utilizing the reSIProcate SIP stack.

As shown in Figure 2.20, the proposed solution has been implemented in two different test-beds (one at the University of Roma Tor Vergata and one at the NEC laboratory), in one field trial with 30 real users realized by BULL Italia in a project commissioned by a customer and in a demonstration being realized for the CNIPA (Italian Center for Information Technology in the Public Administration). In the testbed at University of Rome, we used the Java MMC and MMS, the terminals are Windows XP laptops equipped with Wifi, UMTS cards and Bluetooth dongles, the User agent is Xlite, the SIP proxy is the mjserver available at [33]. In the testbed at NEC we used the same software and hardware for the terminal, while the MMS is the one developed by NEC and the SIP proxy is the SER. In the field trials the MMS is JAVA based, while the terminals are Jasjar i-mate PDAs (with native Wi-Fi and 3G interface) using the Cicero Soft Phone and the proxy is a commercial SIP proxy called Pointercomm SIP commander. In the demonstration at CNIPA the terminal and MMS are the same as in the BULL Italia field trial, but the SIP proxy is the Avaya SES.

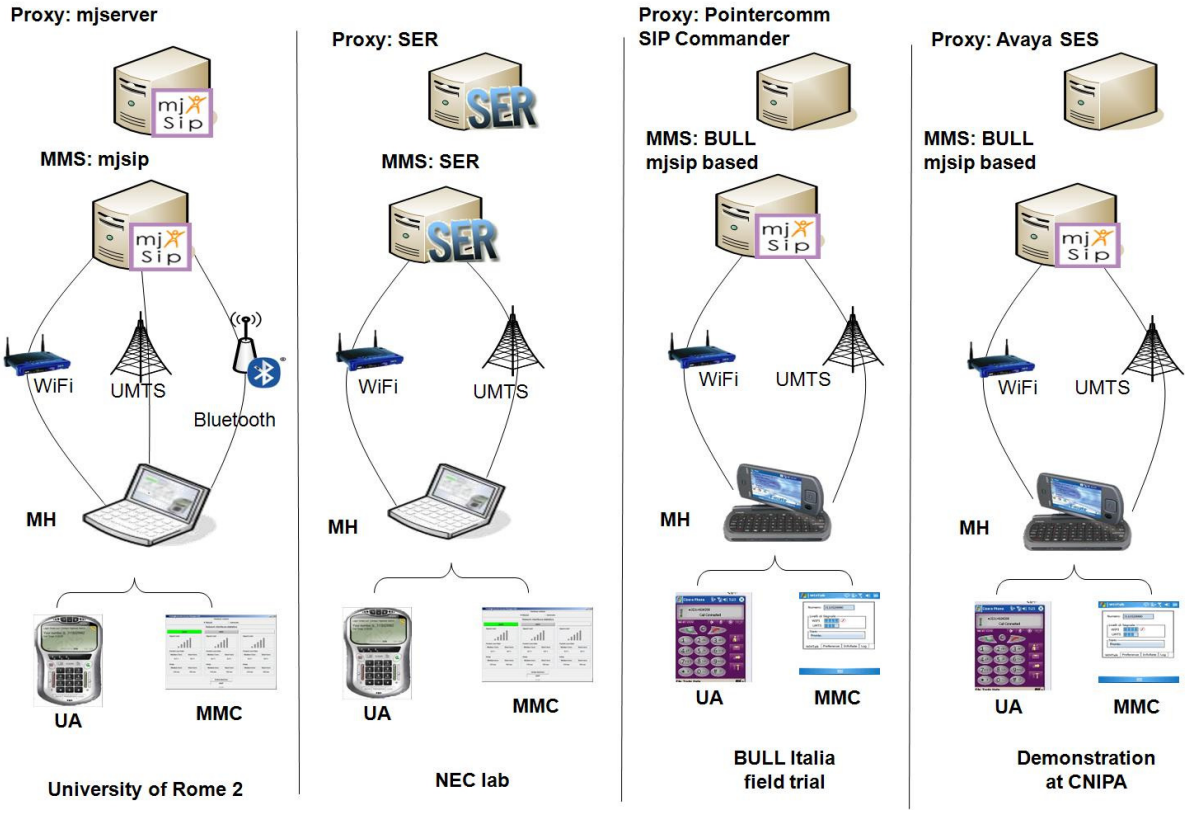


Figure 2.20: Testbeds

3 Evaluation of MMUSE

In this chapter we first evaluate the performance of our implementation of MMUSE with measurements coming out of a some field trials. Then we compare the matching between the solution and the handover requirements defined in section 1.1, also in comparison with other solutions. Finally we evaluate performance of MMUSE and other solutions by using a simple analytical model. In particular we analyze the "disruption time" (defined in 3.3.1.2) for MMUSE, MIPv4 and SIP RE-INVITE mobility in a comparable scenario.

3.1 MMUSE: Handover Performance

We have implemented the proposed solution and realized a testbed across our University campus network (both over WiFi and Bluetooth), a Wifi network connected to an operator's network (Telecom Italia) via ADSL and two different 3G networks (Vodafone and TIM). The testbed layout is shown in Figure 3.1. The Mobile Terminals has been implemented using laptops with Windows XP SP-2 (this version of XP is only required for Bluetooth), the SBC and the SIP Registrar are implemented on a standard PC (both Windows XP and Linux can be used). MMC and MMS have been implemented in Java using (and modifying) the open source MjSip Java SIP stack [33] As SIP User Agent we used the Xlite software client [32]. The laptop is equipped with an internal WiFi card, with a PCMCIA card for 3G access, and a BT dongle compatible with XP SP-2. As WiFi access network we used both an our own Access Point connected to the Campus Fixed Lan, and a WiFi network in our labs which is connected to Telecom Italia backbone. As 3G network we used the Vodafone network and the TIM network. As Bluetooth access network we used a Linux host with a Bluetooth dongle and the open source "BlueZ" Bluetooth stack [36]. The Linux host is configured to bridge the Bluetooth PAN with the fixed Ethernet LAN, so that a client host connecting to the PAN simply gets an IP address valid on the fixed Ethernet LAN.

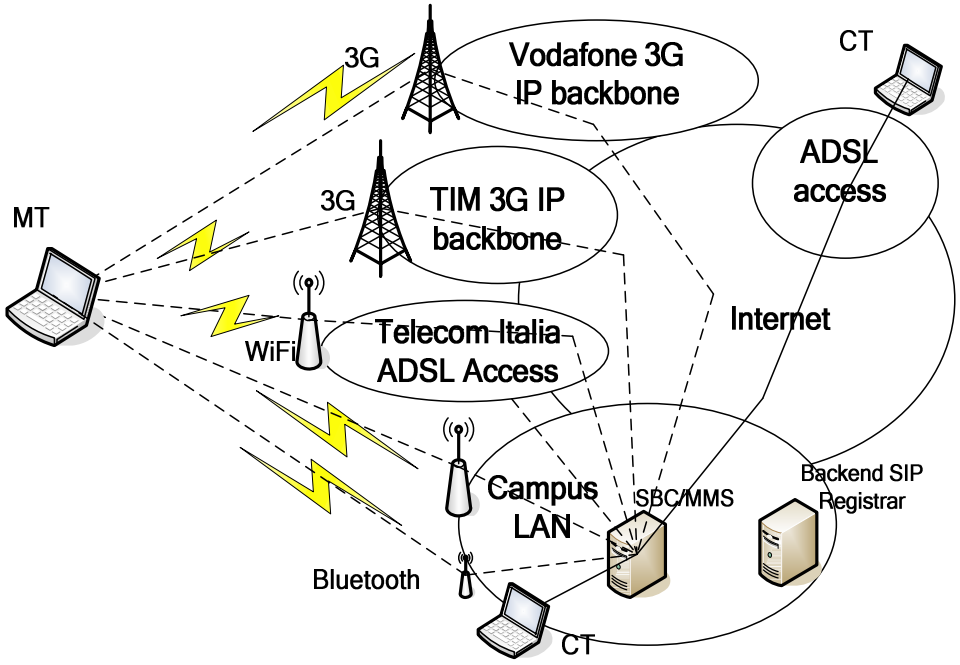


Figure 3.1: Testbed layout

The SBC and SIP Registrar were located in our campus LAN and given a public IP address. As Correspondent Terminals we experimented both a PC in our campus LAN and a PC using an ADSL access.

In the Mobile Terminal, the MMC interacts with the operating system by checking the status of the interfaces with the “ipconfig” command. The MMC offers a simple Graphical User Interface which shows the currently active interfaces and allows to control the handover by choosing the “selected” interface.

No handover decision criteria are implemented in the described testbed. The handover decision is manually provided through the Graphical User Interface of the MMC.

On the testbed we first assessed the performance of the different access networks (in particular the 3G cellular networks) in the support of VoIP application (section 3.1.1) and then the performance of the proposed handover procedure (section 3.1.2). We performed some subjective measurements of the perceived VoIP quality during the HO procedure and we found that the voice impairments are due to the different networks delays experimented by the WLAN (or Bluetooth) and the 3G networks (as shown later in Figure 3.3), while no impairment is perceived making the handover among two networks with the same delay. We are currently working on objective evaluation of voice quality using an approach [27] based upon a reduction of the ITU-T's E-Model [28] to transport level measurable quantities

3.1.1 Evaluation of access network performance

In order to evaluate the performance of different access networks, we have developed a tool named "Throcalc" which is able to evaluate packet loss, Round Trip Time (RTT) and one-way delay jitter with powerful NAT traversal capability. The tool is composed of a client side which runs on a PC (both Windows and Linux OSs are supported) which can be equipped with any network interface and a server side which we run on a PC with public IP address on our university campus network (e.g. on the same host where the SBC/MMS is located). Therefore we are able to evaluate the performance of the "uplink" (from MT to SBC/MMS) and "downlink" (from SBC/MMS to MT) channels over the different wireless network. Note that the performance that we will consider is not only related to the wireless part of the path. For example when evaluating the performance of a 3G network, we include the fixed part of the radio access network, the IP backbone of the 3G operator, the Internet path from the 3G operator up to our campus network and finally the path from the campus network border router up to the SBC/MMS. Anyway this is exactly the path that will be crossed by voice packets that cross the SBC/MMS.

	BlueTooth	WiFi	3G net 1	3G net 2
Average packet loss ratio	0,11%	0,06%	0,79%	0,24%
Maximum packet loss ratio	0,13%	0,13%	2,89%	0,29%

Table 3.1: Packet loss ratio of the different network access

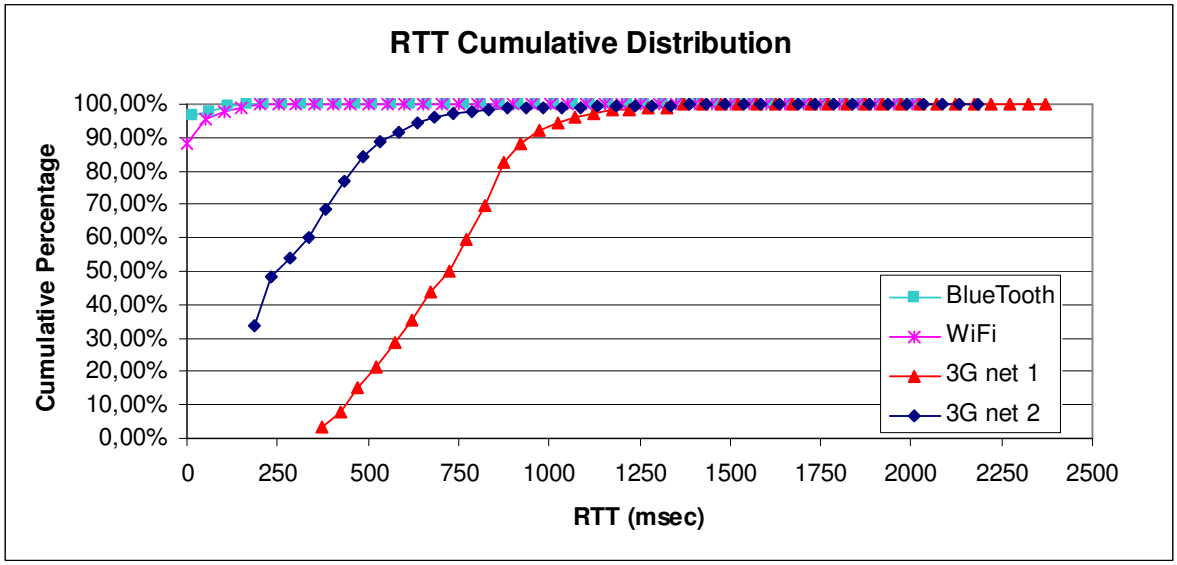


Figure 3.2: Cumulative distribution of RTT for different access network

A more detailed report of the measurement campaign can be found in Appendix A, we only present here the main results. We measured a very good (i.e. low) loss rate using all the different access network. Table 3.1 reports a sample of our loss measurements over the different access network. On the other hand the RTT was not good when using both the 3G networks that we have tested (we recall that the measurements are related to the whole path from MT to SBC, which does not only include portions of 3G network, although we believe that the loss and delays are mainly related to the 3G portion of the path). Figure 3.2 shows the cumulative distribution of RTT for the different access networks. Each distribution is evaluated with 5 different tests of duration 60 seconds repeated with 15 minutes interval during a working hour (e.g. 11 am). The average RTT is in the order of 400 ms and 800 ms for the two networks and even worse is the 95% percentile which is in the order of 600 and 1000ms, resulting in a degradation of voice experience.

3.1.2 Evaluation of handover performance

We analyzed the performance of the handover by capturing the media and signaling packets on the MT and on the SBC, using the Ethereal passive measurement tool [35]. We did not consider the path between the SBC and the Correspondent Terminal, as it does not impact the performance of the handover. The GSM codec at 13 kb/s was used. We have recorded the departure and arrival times of voice packets at the MT and at the SBC. We analyzed both the uplink flow (MT->SBC) and the downlink flow (SBC->MT) and we considered the handovers from WiFi to 3G and vice-versa (in total we have 4 scenarios).

Looking at the 4 graphs in Figure 3.3, in the x axis we put the departure time of packets from the originating interface, while in the y axis we put the arrival time of the packets at the destination interface. As the clocks are not synchronized, the time is relative to the first sent or received packet on the interface and we are not able to measure the absolute “one-way delay”. This is not a problem, as we are interested in the differential delay among arrived packets. For the different scenarios we will discuss: 1) the impact of the difference in the one-way delay between the WiFi and the 3G network during the handover; 2) the handover completion time, i.e. the time elapsed from when the MMC starts the handover procedure and when the procedure is completed and the voice in both directions is flowing on the target interface.

Let use define as Uup and Udn the one way delay for the 3G network in the uplink (MT->SBC) and downlink (SBC->MT) direction. These delay do not only cover the 3G network, but all the path from MT to SBC, crossing the 3G network (see Figure 3.1). Similarly we define Wup and Wdn for the WiFi network. In the performed experiments, the measured round trip time between the MT and the SBC for the 3G access (i.e. Uup+Udn) was in the range of 200 ms (as shown in Figure 3.4), while for the WiFi access (i.e. Wup+Wdn) was in the range of 20-25 ms.

Figure 3.3 reports the results for the 4 scenarios. From the diagrams related to uplink (a and b) we can give an estimate of the difference in the "one-way delay" for the 3G access and for the WiFi access in the "uplink" (i.e. Uup-Wup). As the packets are duplicated, the difference in the y axis between the arrival of the same packet sent on the WiFi and on the 3G interface is the delay difference. It turns out that at the time of our tests, uplink one-way delay experienced in the 3G access is 80 to 110 ms higher than the one experienced in WiFi. A set of packets will arrive from the 3G interface which are the copies of the already arrived packets. This packets, marked with a circle in Figure 3.3-a, will be forwarded and received by the CT as duplicated packets. The duration of this burst of duplicated packets is equal to the difference of the "uplink" one way delay between WiFi and 3G (Uup-Wup). As for the handover completion time, it roughly corresponds to the round trip time on the target interface (3G in this case: Uup+Udn).

We measured 270 ms for the interval between the REGISTER and the 200 OK (i.e. the handover duration as perceived by the MT) in the test shown in Figure 3.3-a. As a confirmation, we can see that in Figure 3.3-a the packets are duplicated from t=12430 ms to t=12700 ms. This duration is almost entirely caused by the round trip time between MT and SBC/MMS. This is confirmed in Figure 3.4 which shows the RTT measured analyzing the traces of RTP packets captured on the MT and on the SBC/MMS for 6 seconds following the handover.

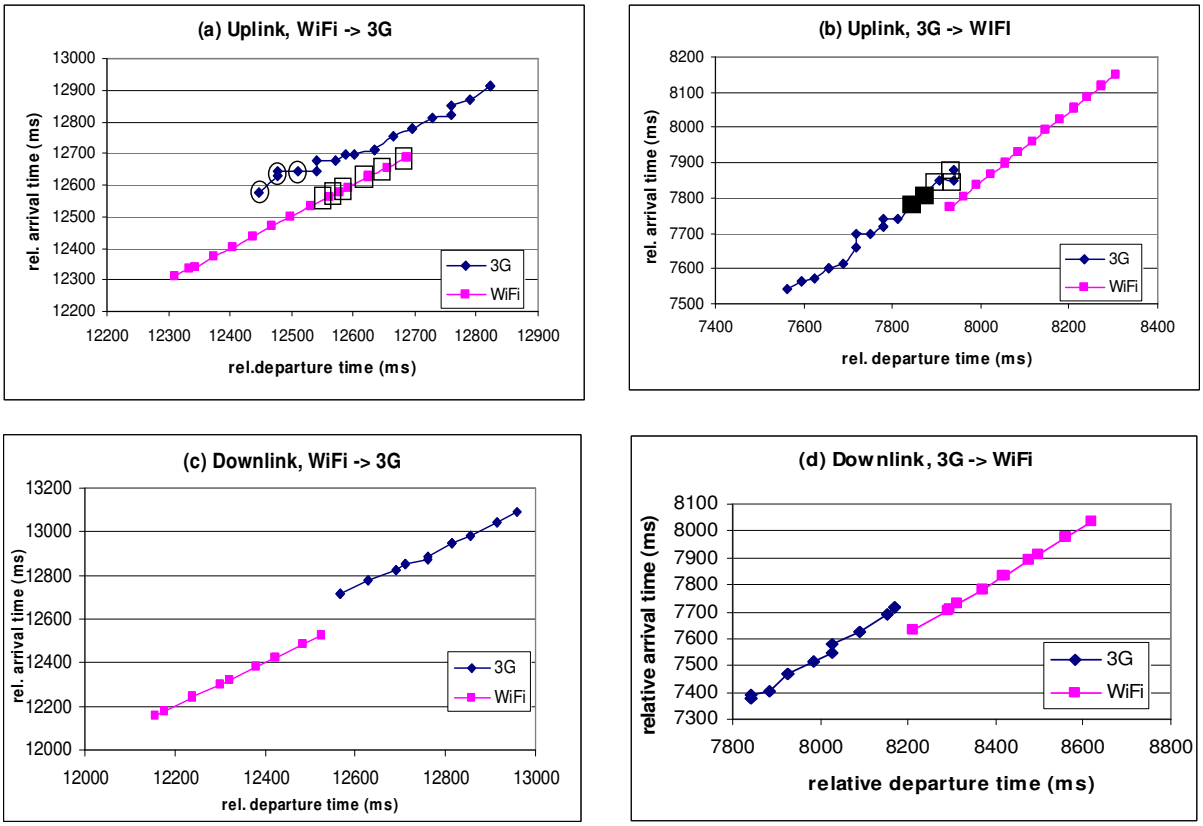


Figure 3.3: RTP arrival patterns during handovers in 4 scenarios

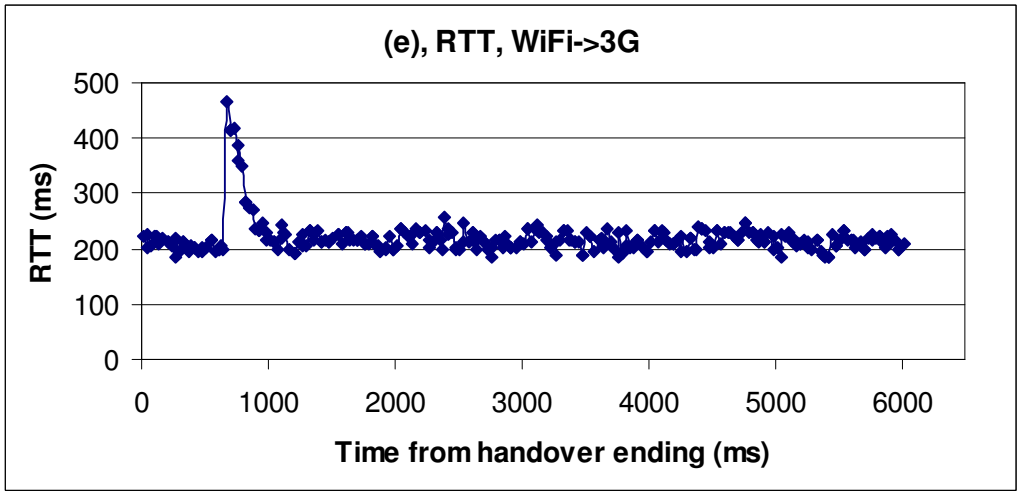


Figure 3.4: Estimation after handover procedure ending (3G network)

In case of the handover from 3G to WiFi (always in the uplink case), the MT sends the SIP REGISTER message on the "faster" WiFi network and starts duplicating the voice packets. When the SBC receives the REGISTER it will start accepting packets sent on the WiFi interface and discarding those sent on the 3G interface (marked with a square in Figure 3.3-b). The first received packets sent on the WiFi interface will have an higher sequence number than the last one received coming from the 3G interface, as the packets sent on the WiFi interface

have “overcome” the ones sent on the 3G interface. A number of packets will be lost, and these packets are marked with the solid square in Figure 3.3-b. The duration of the burst of lost packets is again equal to the difference in the uplink one way delay between 3G and WiFi network. As for the handover completion time, we measured 32 ms for the interval between the REGISTER and the 200 OK in the test shown in Figure 3.3-b. In fact, the packets are duplicated from t=7906 ms to t=7938 ms. Coming to the downlink flows, let us consider the handover from WiFi to 3G (Figure 3.3-c). The SBC will stop sending packets towards the WiFi network and start sending them towards the 3G network when the REGISTER message is received. The first packet sent towards the 3G network will experience an additional delay equal to the difference in the “downlink” one way delay between 3G and WiFi network (Udn-Wdn) which is in the order of 200 ms in Figure 3.3-c. The gap shown does not represent a loss of some packets, it only shows a delay between the reception of the last packet sent on the wifi interface and the reception of the first packet sent on the 3G interface.

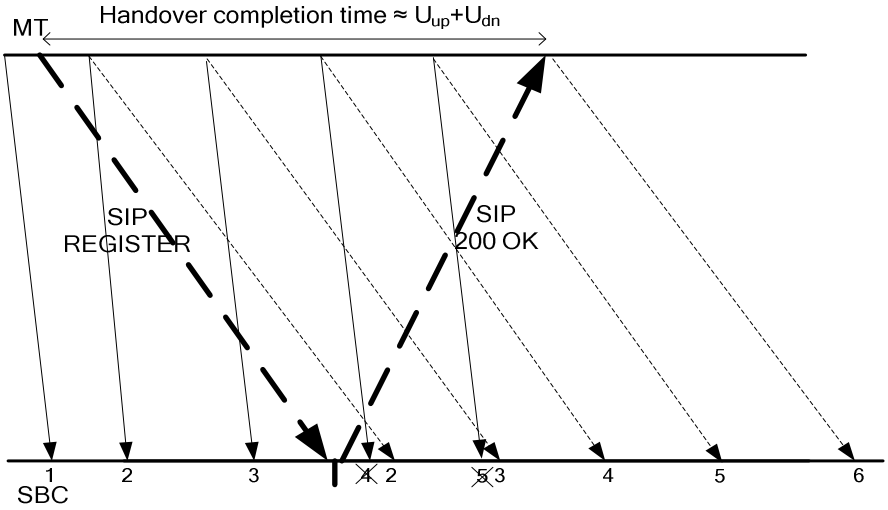


Figure 3.5: Temporal diagram for WiFi -> 3G handover

Finally, let us consider the handover from 3G to WiFi for the downlink flow (Figure 3.3-d). As soon as the REGISTER message is received by the SBC the packets will be sent towards the WiFi interface and will arrive at the MT in advance with respect to packets with lower sequence number previously sent towards the 3G interface. The duration of the advance is equal to the difference in one-way delay (in the order of 200ms in Figure 3.3-d. Note that no packets are lost in this handover, the gap in Figure 3.3-d represents the timing advance of the first packets sent on the wifi interface that experience lower delay and arrive before the last packets sent on the 3G interface.

Similarly to what we have done in the uplink, from the data reported in Figure 3.3-c and d we can evaluate the difference in one-way delay for the downlink Udn-Wdn. In our test we measured that 3G one-way downlink delay was from 80 to 110 ms higher than WiFi. The results show that the different delay between the WiFi and 3G network is a critical factor. If the differential delay is reasonably low the voice decoder is able to hide the handoff. In our tests, where Uup-Wup and Udn-Wdn are in the order of 110 ms, the handovers are not perceived.

It is interesting to compare the results shown in Figure 3.3 with the corresponding measurements without using the keep-alive mechanism introduced in section 2.4.3. As we can see in Figure 3.3, which reports the uplink measurement for the WiFi to 3G handover, the initial differential delay between the 3G and WiFi is in the order of 2,8 s. Correspondingly, we have that the duration of the handover (during which all packets are duplicated) is in the order of 3 s. This is due to the fact that starting to transmit over a 3G interface requires a considerable amount of time. Just for comparison, we have reported the diagram of Figure 3.3-a in an arbitrary position in the left part of Figure 3.6.

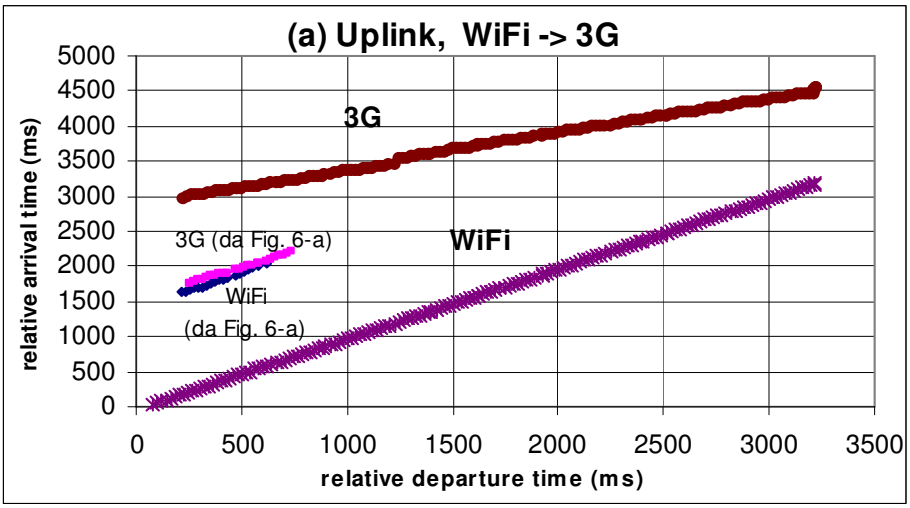


Figure 3.6: RTP arrival pattern without keep-alive on the unselected 3G interface

It is possible to appreciate the difference in terms of handover duration and of the distance between 3G and WiFi packet arrival time. The conclusion is that the keep-alive mechanism is needed to support seamless handover. The results shown in Figure 3.3 consider the favorable case in which both interfaces remain active during the handover. It can happen that the old interface goes down suddenly and does not allow to transmit packets during the handover. In our solution, the uplink flows are not affected, as the MT

starts sending packets on the new interface immediately. On the contrary, the downlink flows are affected, as the SBC will start transmitting packets towards the new interface only after receiving the handover request from the MT. We have analyzed this case with temporal diagrams similar to the one shown in Figure 3.5 and we have repeated the measurements of handover performance. The full results are not shown here for space constraints, anyway we have found theoretically and measured from the testbed that on the downlink flow we have an impairment in the order of Uup + Wdn for the handover 3G -> WiFi and in the order of Wup + Uup for the handover WiFi -> 3G.

Similarly to what we have done for WiFi and 3G networks, we evaluated the performance of our handover mechanism, switching from Bluetooth to 3G network and vice versa. The performances of the BT network are very similar to the behavior of WiFi network as shown in Table 3.1 and in Figure 3.2. To support this statement, in Appendix A we report all the uplink and downlink measurement results for the handover between Bluetooth and 3G In one sample measurement reported, we obtained 265 ms for the interval between the REGISTER and the 200 OK (i.e. the handover duration as perceived by the MT).

The reported results show that the impairments in the handover procedure are due to the intrinsic RTT of the “target” network and to the differential one way-delays between the origin and the target network.

3.2 Matching Criteria

In this subsection, we check if other Mobility Management solutions discussed in section 1.2 and our proposed solution match the requirements that we listed in section 1.1. As we can see from Table 3.2 MMUSE is able to fulfill all requirements, which is not true for the other approaches.

	Mobile IP	SIP Re-Invites	MMUSE	SIP based as in [14]
1 Handover to be as fast as possible	OK (when using Fast HO extension in MIPv6; still a proposal for MIPv4)	Not optimized	OK (depending on MMS location)	Not Optimized
2 Forward handover	OK (when using Fast HO extension with bi-casting in IPv6; still a proposal for MIPv4)	OK	OK	OK
3 Nat traversal	Require MIPv4 extension defined in RFC 3519	Can be complex	OK	Not considered, it can be complex
4 No need of support in CT	OK (however, support is needed with route optimization)	NO, support is needed	OK	NO, support is needed
5 No need to change UAs in MT	OK	NO, UAs needs to be changed	OK	NO, UAs needs to be changed
6 No need of support in AN	NO, support may be needed (FA in IPv4, MAP in Hierarchical MIPv6)	OK	OK	OK
7 Handover can be provided w/o operators' support	NO	OK	OK	OK
8 Hiding MT location and movements	OK (with reverse routing and/or Hierarchical MIPv6)	NO	OK	NO
9 Interworking with SIP Personal Mobility	OK	Could be realized, but a two level naming for Mobile Terminals / Users should be defined	OK	OK

Table 3.2: Mobility Management requirements matching

3.3 MMUSE vs. Other solutions: Performance evaluation

As shown in section 1.2 there are a large number of different mobility management solutions in the literature, operating at different protocol stack layer.

More often the solutions are proposed and discussed only at the architectural level, without an implementation or a performance analysis. Some works include an implementation and/or try also to make performance consideration and comparison among different solutions. In this section we will discuss some methodological aspects about how the solutions should be evaluated and compared. We will try to identify the set of reference scenarios and the set of performance metrics that should be evaluated.

A solution for mobility management needs to include the procedures to keep track of device movements, while it does not necessarily need to provide support for handover of active sessions. However the handover capability is very important to provide a seamless user experience, and it is the most interesting benchmark for comparing different solutions. Therefore we will only consider solutions that provide the handover capability and we will compare these solutions in their handover performance.

3.3.1 Methodology for performance evaluation

As stated earlier, the solutions for mobility management are often presented without discussing the performance related aspects and without providing results from implementation. When some implementation is discussed, it typically shows only some "success" in a comparable scenario where handover signalling worked fine. We propose a methodology for comparing the performance of different handover solutions, including the proper evaluation of their behavior in presence of packet loss events. In principle, the comparison could be performed with practical experiments (see for example [14], [16], [17]), with simulation or with analytical methods (see for example [18], [19]). We are interested in the analytical methods, and in particular we would like to extend the already proposed approaches by taking into account the loss rate that can affect the packets of the data and signalling flows involved in the mobility management procedures.

In the following subsections, we first classify the handover scenarios, then we define the performance metrics.

3.3.1.1 Handover Scenarios

With the help of Table 3.3, we classify the handover scenarios as follows. A device can have one single interface on a specific technology (e.g. WiFi) or more interfaces on the same or on different technologies (e.g. two WiFi interfaces, or one WiFi and a 3G interface).

	Handover types
One Single Interface	S1: "Sub-IP" S2: "Explicit HO"
More interfaces	M1: "Two-active-Ifs" M2: "Two-Ifs-one-breaks" M3: "One-active-IF"

Table 3.3: Classification of handover scenarios

In case of one single interface, we can have two cases which we name S1 ("Sub-IP") and S2 ("Explicit HO"). A "Sub-IP" handover is not perceived by the device at the IP level and above. For example WiFi handovers among access points in the same "ESS" (Extended Service Set) are fully handled at layer-2, so that the IP address of the terminal is not changed. Similarly a 3G interface will handle handovers in the cellular network across base station, without changing IP address. On the other hand the device may need to handle an "Explicit Handover" even on a single interface. This may be the case on a WiFi interface when going out from the coverage area of a set of access points operated by an organization and connecting to an access point operated by a different organization.

When (at least) two interfaces are involved, we envisage three different handover cases denoted as M1, M2 and M3. In the handover of type M1, the terminal is able to communicate over the two interfaces and the two interfaces remains active even during the handover execution. In the handover of type M2: the terminal is able to communicate over the two interfaces, but the communication on the "old" interface suddenly breaks, so that the communication needs only to be moved on the new interface. Finally, in the handover type M3 the terminal is connected on one interface both before and after the handover. In this case the communication on the new interface needs to be established "from scratch" as the first step in the handover execution. The three arrows in Figure 3.7 represent the movement of a mobile terminal that performs the three different types of handover: in the M1 case the terminal is under double coverage of the two Access Networks (AN), in the M2 case the terminal is moving from an area of double coverage to an area of single coverage, in

the M3 case the terminal moves from an area of coverage of AN1 to an area of coverage of AN2.

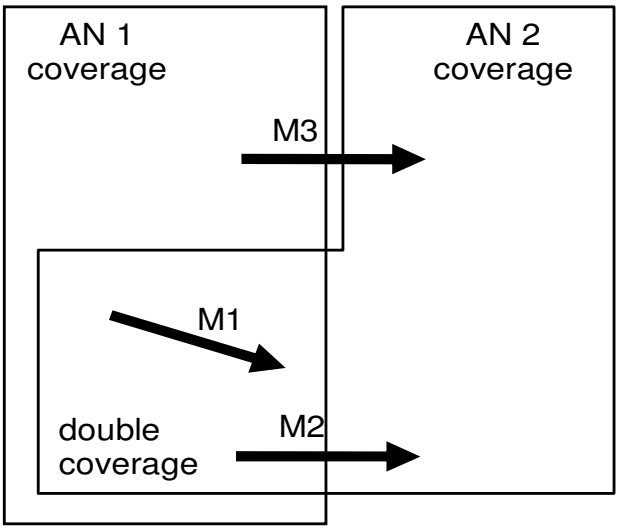


Figure 3.7: Handover scenarios with 2 different access network technologies

3.3.1.2 Performance metrics

Our main performance metric is the "disruption time" *dt* perceived by the two users during the handover procedure. It can be different in the two directions, assuming that a mobile node is communicating with a fixed correspondent node. Therefore we define *dtcn* as the disruption time for the uplink flows (from mobile node to correspondent node) as perceived by the correspondent node and *dtmn* as the disruption time for the downlink flows (from correspondent node to mobile node) as perceived by the mobile node. When we want to evaluate a specific mechanism for handover we will first classify the handover types as defined above and for each supported type Mi we will consider the sequence of messages to be exchanged. Then we can evaluate *dt(Mi)* as a function of the network delays among the involved entities and of the processing delays at the involved entities. In particular we define as the disruption time in case of a successful handover procedure (e.g. where no messages are lost) and as the average disruption time assuming a loss probability for the message delivery.

The second considered performance metric is the number of mobility management signalling packets exchanged denoted as M. Similarly, we define as the number of exchanged signalling packets in case of a successful handover of type Mi and as the average number of exchanged signalling packets assuming a loss probability for message delivery.

We note that if we assign a weight wi to the different handover types Mi, we could evaluate both the disruption time and the number of exchanged signalling packets as a weighted sum of *dt(Mi)* and *M(Mi)* respectively.

3.3.2 Mobility Management Mechanisms

In this section we introduce the mobility management mechanisms we compare in this document: Mobile IP, SIP re-INVITE Method, SIP MMUSE.

Using Mobile IP, the mobile node can be reached with the same IP address, regardless of its movements. Mobile IPv4 (MIPv4) [9] foresees that packets incoming to the mobile node are sent to an entity called “Home Agent” (HA), tunneled to the so called “Foreign Agent” (FA) and then delivered to the mobile node. Making an handover with “canonical” MIPv4 means moving from an “old” FA to a “new” FA and involves the HA as well. A lot of solutions have been proposed to improve the handover performance of Mobile IPv4

The SIP re-INVITE solution is the handover solution foreseen in SIP standards [8], [13]. It only relies on the capability of the mobile nodes. A terminal that is making the handover sends a request (a SIP re-INVITE message) to its correspondent, providing the new addresses for re-establishing the media flows.

The MMUSE (Mobility Management Using SIP Extension) handover solution has been described in chapter 2. It relies on an intermediate entity (the MMS, Mobility Management Server) to handle the handover. Under this solution, no support is needed from the correspondent terminal in order to perform the handover.

3.3.3 Evaluated Scenarios

The reference network scenario for our analysis is reported in Figure 3.8. With reference to the handover scenarios described in section 3.3.1.1 above, we are going to analyze two of the cases with more interfaces: case M2 (“Two-IFs-one-breaks”) and case M3 (“One-active-IF”). We define: th: as the delay between the Mobile Node (MN) and its Home Network (HN); ts: as the delay between the MN and a Foreign Agent (nFA or oFA); tmc: as the delay between the MN and the Correspondent Node (CN); thc: as the delay between the CN and the HN.

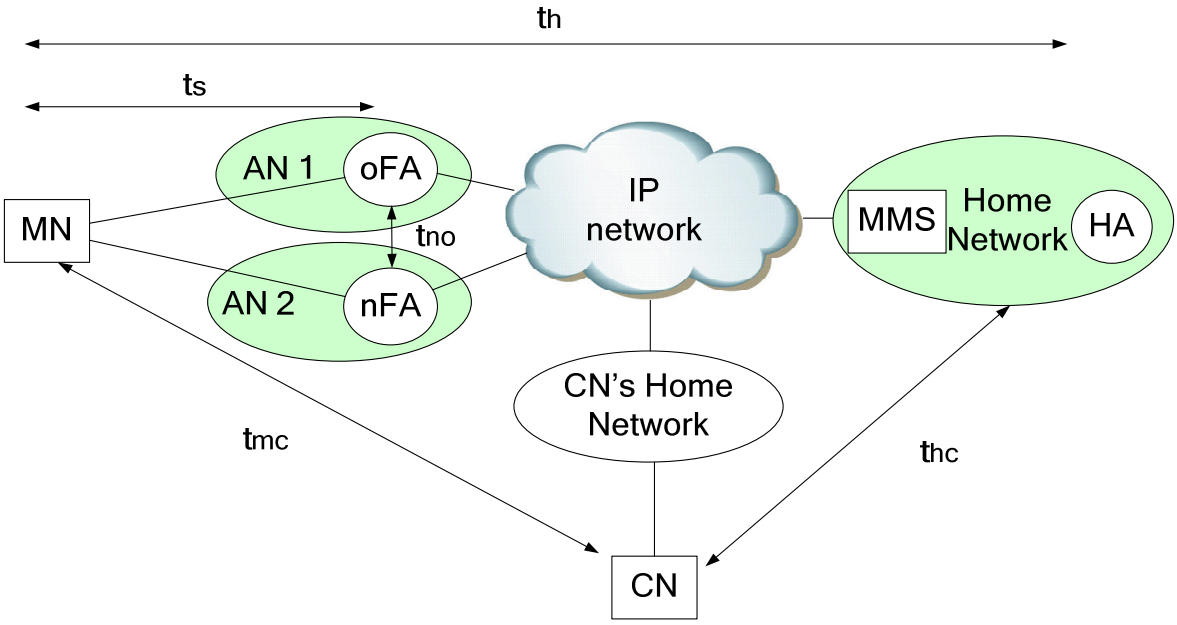


Figure 3.8: The Reference Scenario

3.3.3.1 Case M2 ("Two-IFs-one-breaks")

In the MIPv4 case, we assume the mobile node already knows the new CoA offer from the nFA, so it only needs to activate the registration procedure. In the SIP cases, the MN has 2 IP addresses obtained with 2 different DHCP requests (one for each interface). The handover procedure is simply activated by sending a re-Invite (in the SIP re-INVITE solution) or a Handover Registration request (in the MMUSE solution). Figure 3.9 shows the signalling flows of those procedures.

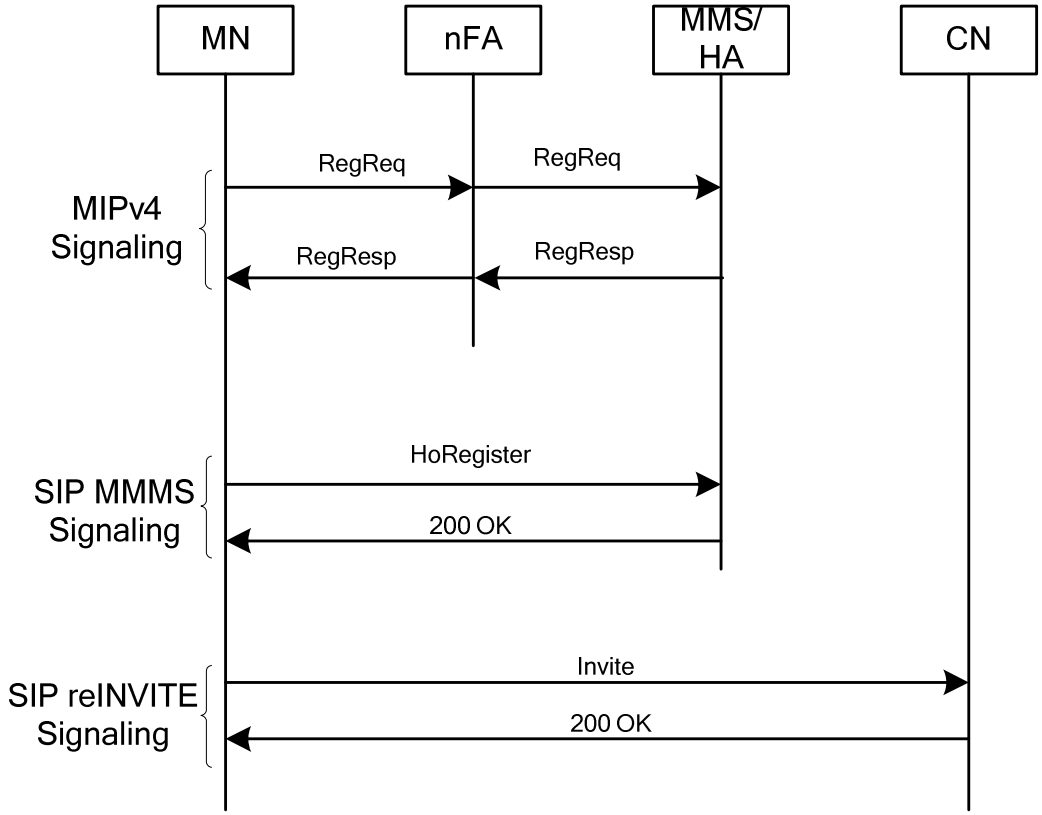


Figure 3.9: Signaling Flows of Mobility Management Mechanisms in M2

3.3.3.2 Case M3 ("One-active-IF")

If the mobile node has only one interface with a valid IP address when this interface goes down, it must first acquire a new IP address on the new interface:

- in MIP [9] the MN must send a Proxy Rt solicitation to FA and waiting a response with his CoA. This procedure has a duration of 2 ts.

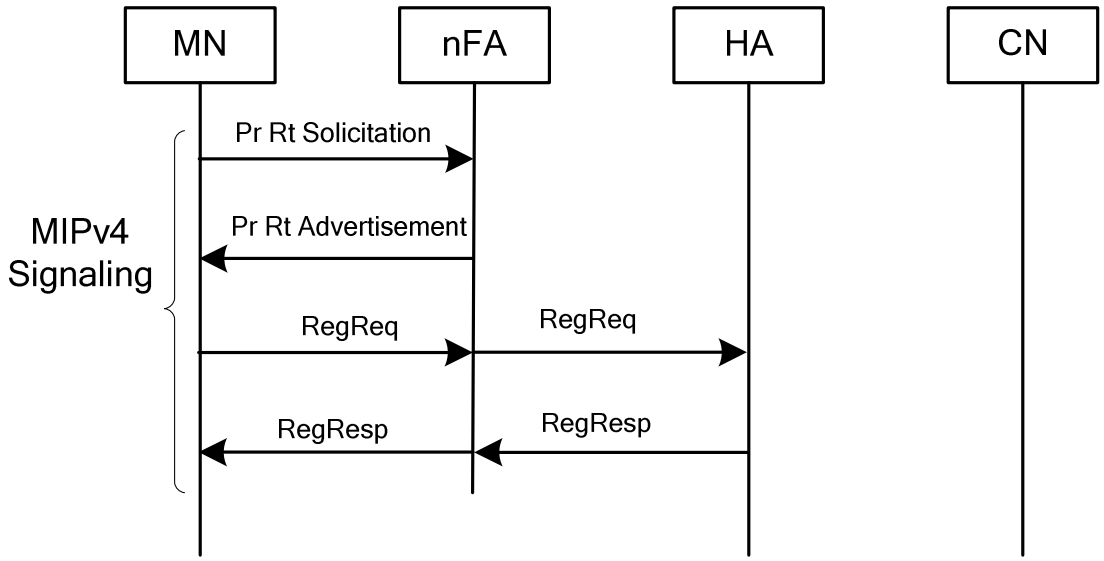


Figure 3.10: MIPv4 Signalling Flows of Mobility Management Mechanisms in M3

- in SIP procedures we need another protocol (e.g. DHCP) to obtain an IP address, with a duration of 4 ts.

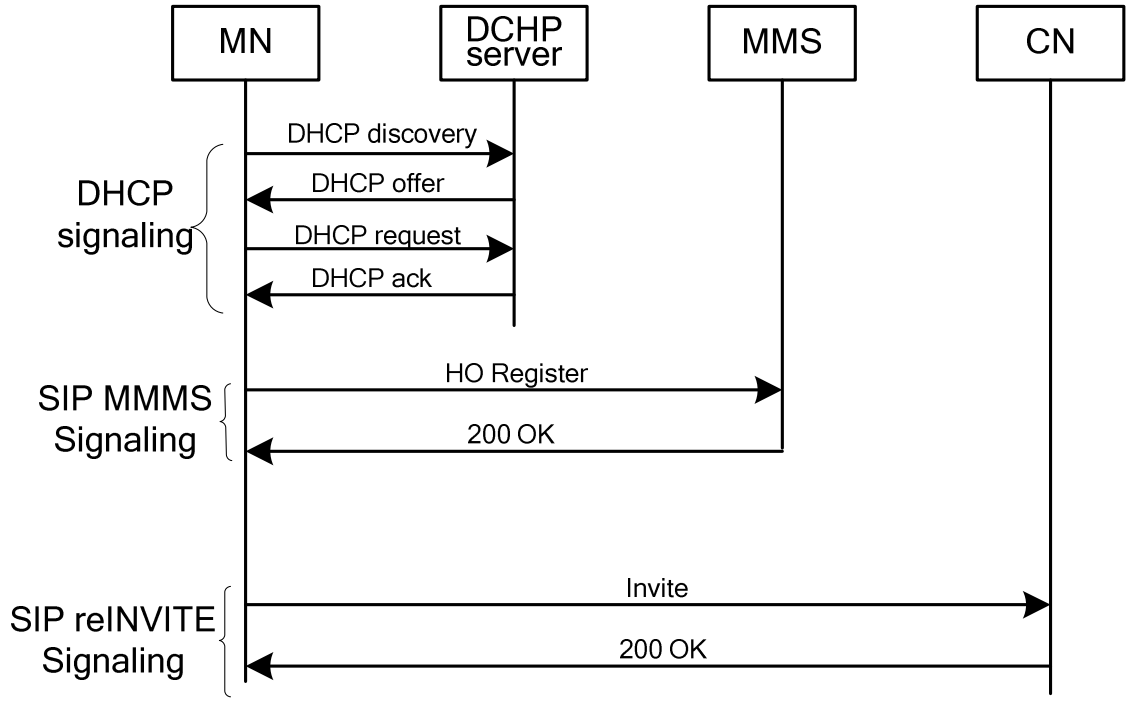


Figure 3.11: SIP Signalling Flows of Mobility Management Mechanisms in M3

3.3.4 Handover Performance Evaluation (No Failure Case)

3.3.4.1 MIPv4

The MIPv4 procedure in the M2 case ("Two-IFs-one-breaks") is the following:

When the mobile node recognizes that the active interface has lost connectivity, it sends a Registration Request to the HA through the nFA. This message contains the CoA of the nFA (we suppose that MN has previously discovered this address).

After th time the HA receives this request and sends a Registration Response to MN. Simultaneously it forwards the media datagram received from CN to the MN. The MN receives the first media packet at:

$$dt_{MN} = 2t_h \quad (1)$$

after the break.

When the MN receives the Registration Response, it can send the first media packet through a new network, so the CN will receive this packet at:

$$dt_{CN} = 2t_h + t_{mc} \quad (2)$$

3.3.4.2 SIP re-INVITE

In the SIP re-INVITE solution, the MN sends a SIP re-INVITE message to CN, communicating the new IP address. When the CN receives this message (tmc), it sends a 200 OK message and immediately sends the first media packet. So the MN receives the first media packets on the new interface at 2 tmc. The CN stops receiving media packets through the old interface at tmc after the start of the handover procedure, while it receives the first media packet through the new interface at 3 tmc time. So the CN has a disruption time of 2 tmc.

3.3.4.3 MMUSE-solution

In the MMUSE-solution, when the old interface breaks down, the MN sends the SIP HO-Reg and the media packets to the MMS (th) on the new interface. Therefore, the CN does not perceive any disruption.

When the MMS receives the Register, it sends the 200 OK and starts forwarding media packets that it receives from the MN to the CN and vice-versa. Note that the CN does not know the MN's status so it continues to send the media packets to MMS. The MN receives the first packet on the new interface at th after the old interface was lost.

In the case M3 ("One-active-IF") (case M3) we must add to all the disruption times the time needed to obtain a valid IP address on the new interface. This time is 2ts for MIP and 4ts for SIP procedure.

3.3.4.4 Comparison of handover Procedures

The following table shows the disruption time in the previously described scenarios.

	M2 case ("Two-IFs-one-breaks")	**M3 case ("One-active-IF")**
MIPv4	$\begin{cases} dt_{CN} = 2t_h + t_{mc} \\ dt_{MN} = 2t_h \end{cases}$	$\begin{cases} dt_{CN} = 2t_s + 2t_h + t_{mc} \\ dt_{MN} = 2t_s + 2t_h \end{cases}$
SIP re-Invite	$\begin{cases} dt_{CN} = 2t_{mc} \\ dt_{MN} = 2t_{mc} \end{cases}$	$\begin{cases} dt_{CN} = 4t_s + 2t_{mc} \\ dt_{MN} = 4t_s + 2t_{mc} \end{cases}$
MMUSE	$\begin{cases} dt_{CN} = 0 \\ dt_{MN} = 2t_h \end{cases}$	$\begin{cases} dt_{CN} = 4t_s \\ dt_{MN} = 4t_s + 2t_h \end{cases}$

Table 3.4: Disruption time in handover procedures

In order to compare the various handover mechanisms we plot the disruption time as function of the delay between MN and HN (Figure 3.12) and as function of the delay between the MN and the CN (Figure 3.13) in the M2 case. Figure 3.14 and Figure 3.15 represent the same analysis for the M3 case. In all the figures we have fixed the values: ts=10ms tno=5ms. Moreover in Figure 3.12 and Figure 3.13 we have tmc=25 ms, while in Figure 3.14 and Figure 3.15 we have th=12 ms. This was same configuration as studied in [18] and [19]. Note that in each figures we are considering the maximum disruption time between dtMN and dtCN shown in Table 3.4.

In Figure 3.12 we notice that both MIPv4 and MMUSE grows linearly with the delay between MN and Home Network, with a slope of 2, but MIPv4 needs to add also the delay between MN and CN. The SIP-reINVITE solution is obviously not dependent on the delay between MN and HN, as it is only handled by the two involved terminals.

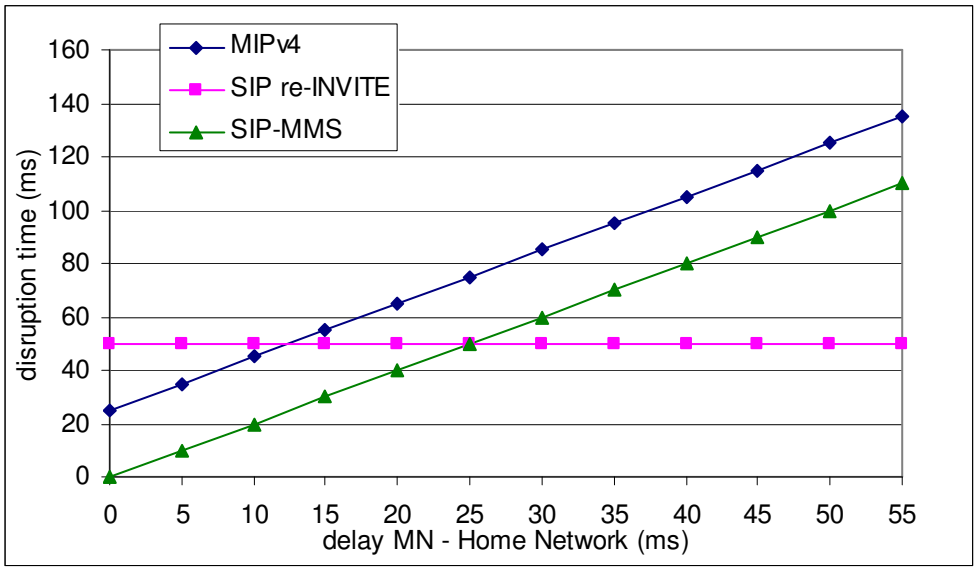


Figure 3.12: Disruption as function of delay MN-HN in the M2 case

Figure 3.13 shows the disruption time as function as to delay MN-CN. The MIPv4 and SIP-reINVITE solutions grow linearly with respect to this delay, but with a slope respectively 1 and 2. The MMUSE solution is not dependent on this delay and it has a constant value.

Figure 3.14 and Figure 3.15 show respectively the same situations in Figure 3.12 and Figure 3.13 but they add the delay necessary to discover the valid IP address for the new IF. So the slope of the disruption time is not varied but the all the delays are increased.

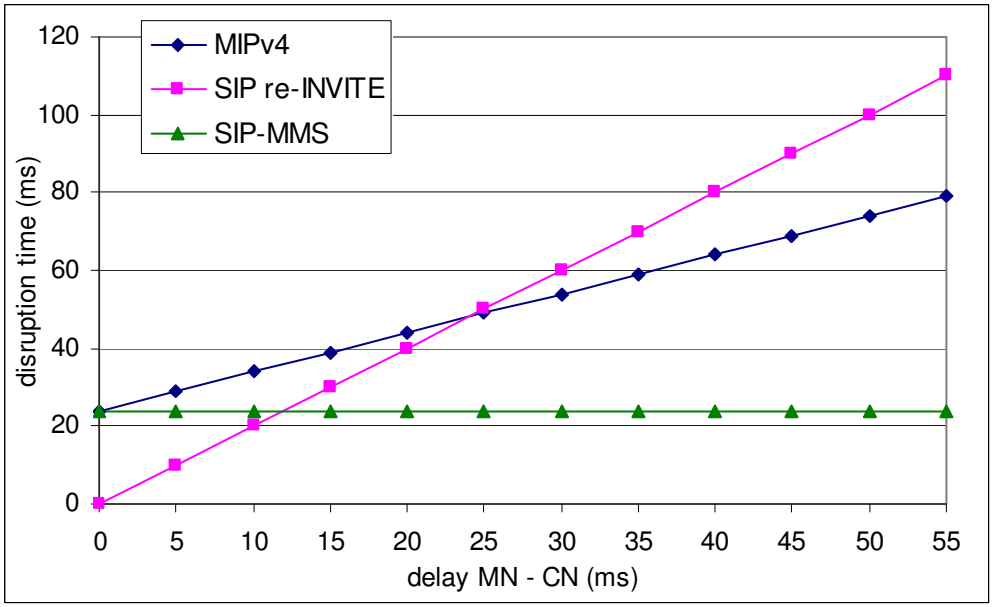


Figure 3.13: Disruption time as function of delay MN-CN in the M2 case

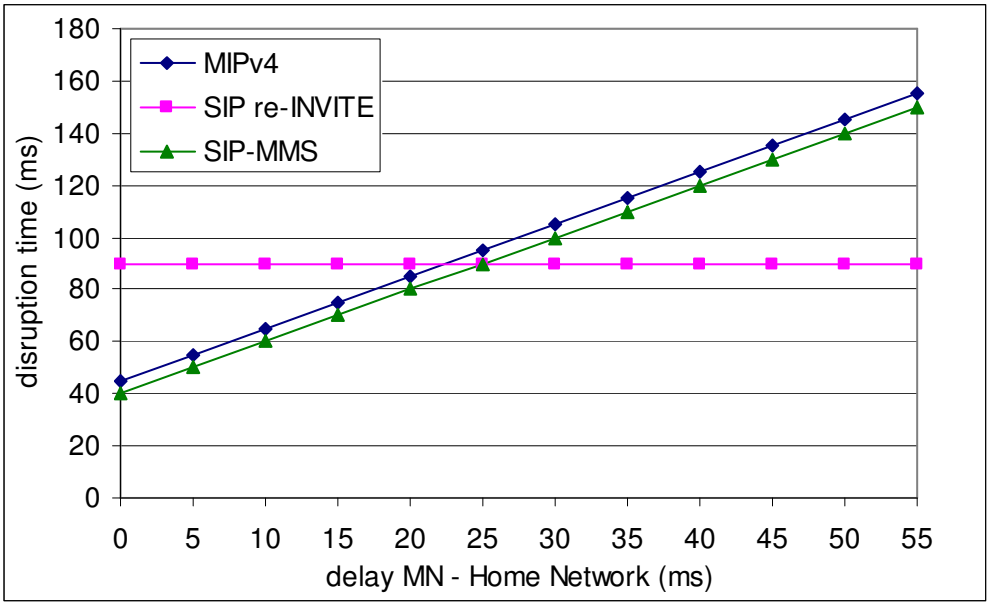


Figure 3.14: Disruption time as function of delay MN-HN in M3 case

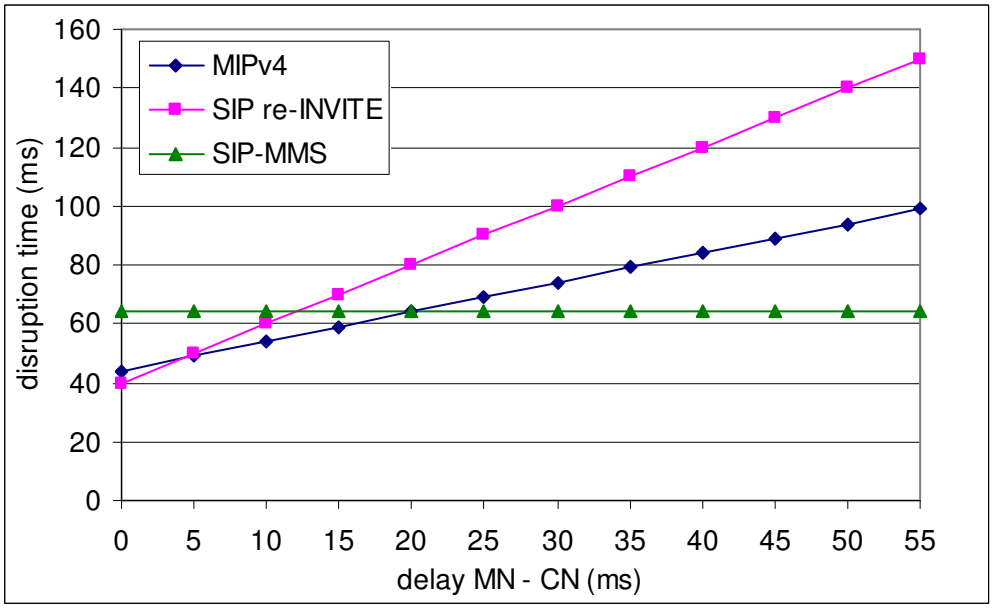


Figure 3.15: Disruption time as function of delay MN-CN in M3 case

3.3.5 Handover Performance Evaluation: Failure Case

As for the performance analysis including failure cases, we only consider the M2 case for space constraint. We can have two types of failure case:

4) the “server-entity” does not receive a Request Message sent by the “client-entity”;
5) the “client-entity” does not receive the Reply Message sent by the “server-entity”

3.3.5.1 MIPv4

When a Registration Request or Registration Reply is lost, the MN waits for a timeout and then re-transmits the Request. The MIPv4 RFC [9] does not define a fixed value to this timeout, it define only a lower bound that is 1 s. We can consider this time as:

$$R_{mip}^{0}[ms] = 2RTT + 100 \geq 1000 \quad (3)$$

For each next retransmission we must add twice this time, so if we define dt(0) as the disruption time evaluated in IP re-INVITE solution, , we can have:

$$dt(n) = 2t_h + t_{mc} + R_{mip}^{0} \sum_{n=0}^{N} 2^n \quad (4)$$

3.3.5.2 SIP re-Invite

The SIP RFC [8] defines that the timeout at the first retransmission must be:

$$R_{sip}^{0} = T_1 \quad (5)$$

where T1 is a value that estimates the RTT of the network ([8] suggests that T1=500 ms). Two different retransmission mechanisms are defined. If we are retransmitting a Request for an “Invite transaction” the timeout is:

$$R_{SIP}^{0} = \sum_{n=0}^{N} 2^n T_1 \leq 64T_1 \quad (6)$$

where n is the number of attempts. In case of a “Not-Invite Transaction” the timeout is:

$$R_{n\_inv} = \sum_{n=0}^{N} \min(2^n T_1, T_2) \leq 64T_1 \quad (7)$$

where T2 should be set to T2=4s

In the SIP re-INVITE solution, we must use the Rinv timeout, so we have this disruption time:

$$dt(n) = t_h + R_{sip}^{0} \sum_{n=0}^{7} 2^n \quad (8)$$

3.3.5.3 MMUSE solution

In the MMUSE solution, if the request is lost, we must send another request to MMS. Since a REGISTER message initiates a “not-invite transaction” the retransmission mechanism follows 2=4s

In the SIP re-INVI so we have the following disruption time:

$$dt_{mmuse\_req}(n) = 2t_{mc} + R_{sip}^{0} \sum_{n=0}^{11} \min(2^{n} T_1, T_2) \quad (9)$$

The MMS starts sending media packets when it receives the REGISTER message. Even if the 200 OK response is lost, the handover procedure can be considered finished when the first media packet is received on the new interface. Therefore in this case the disruption time will become:

$$dt_{mmuse\_resp}(n) = t_h + n\Delta_{rtp} \quad (10)$$

where n means the number of RTP packets that are not received by the MN e the inter-departure time of RTP packets.

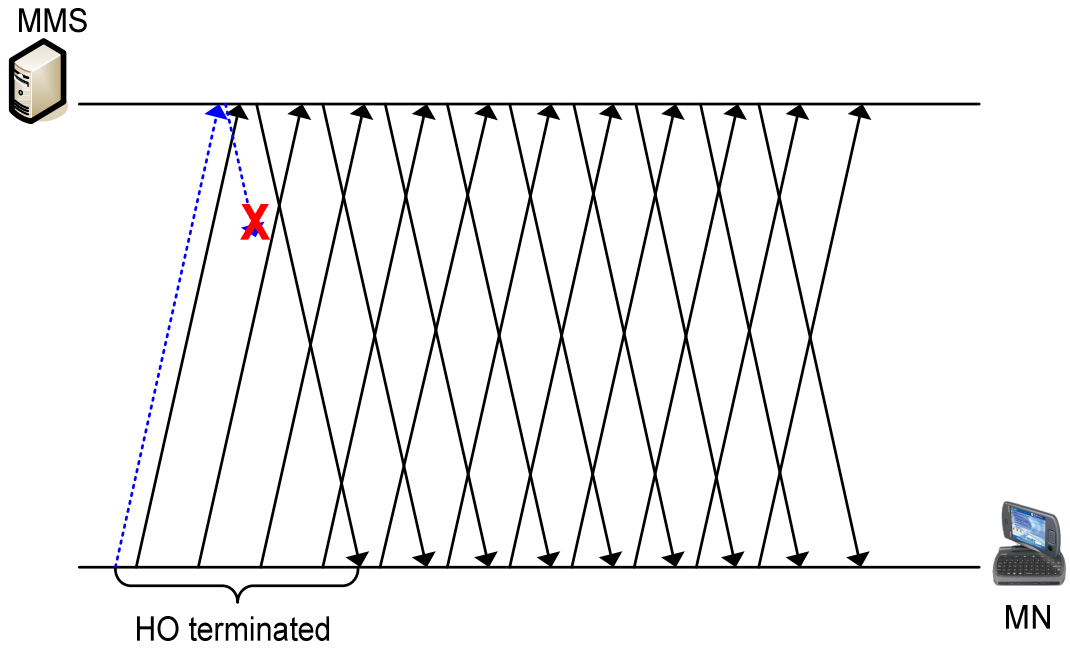


Figure 3.16: MMUSE solution when the responses are not received

In this way the information that the MMS has received the answer is not lost and due to the frequent transmission of RTP packets we are able to significantly reduce the disruption time, with respect to a traditional mechanism with retransmission of requests.

We note that the typical timing of request retransmission mechanism is not very well suited to handover of real-time connections, as the default retransmission timers are too long. Therefore we have proposed in our MMS based solution a faster retransmission mechanism ("MMS-fast") by changing the value of timers (T1=50 ms, T2= 200 ms). We note that this modification is compliant with SIP standards (which only suggest values for T1 and

T2) and that it is very easy for us to apply this timer reduction only to handover register procedures. Of course the drawback of reducing retransmission timer is to increase the signalling load on the network and on servers due to “useless” retransmissions. We define a retransmission is “useless” if it is sent in the time before receiving the response to the initial request and there is no loss in the network. Figure 3.17 shows the number of useless retransmission as a function of the RTT between Mobile Node and Home Network.

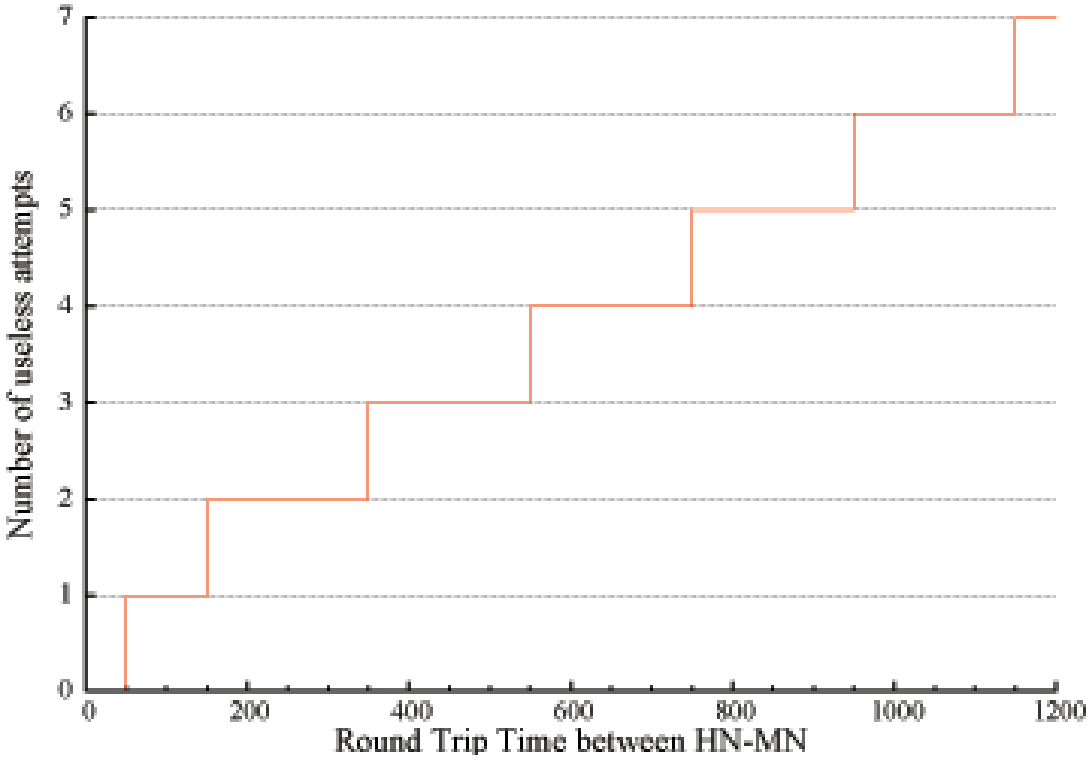


Figure 3.17: Useless retransmissions vs. MN-HN RTT

Figure 3.18 shows disruption time as a function of needed retransmissions. We can see that using fast-MMS the disruption time is less than 500 ms even if three retransmissions are needed.

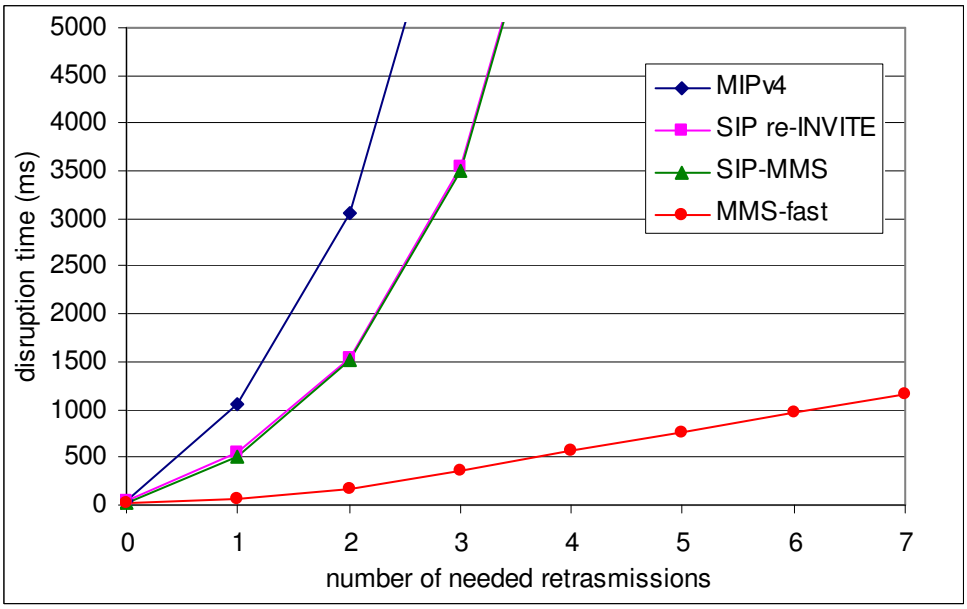


Figure 3.18: Disruption time in the failure case

Let us ph and pmc be respectively the loss probability on the network paths MN-MMS and MN-CN. If we assume that ph=pmc=p we can approximate the average disruption time as:

$$\overline{dt} = \sum_{n=1}^{N_{\max}-1} (\mathrm{dt}(n-1)\cdot p^{n}(1-p)) + \mathrm{dt}(N_{\max}-1)\,p^{N_{\max}} \qquad (11)$$

where Nmax is the Maximum number of retransmissions consents from the handover procedure.

Figure 3.19 plots the average disruption time vs. the loss probability p when Nmax=4.

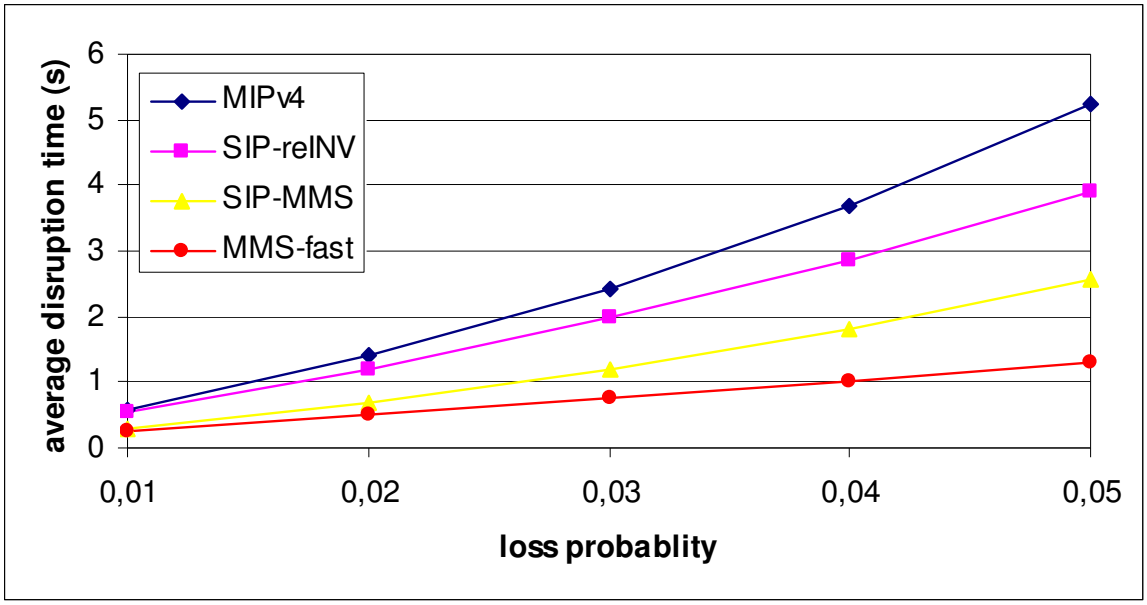


Figure 3.19: Average Disruption Time

4 MMUSE Improvements

In the definition of the Mobility Management solution described in chapter 2, scalability issue was not directly addressed. The solution shown in chapter 2 describes the signaling procedure simply assuming a single MMS that carries over all the traffic (signaling and media) of each Mobile Terminal. On the other hand a scalable solution would of course foresee:

6) a set of MMS that can cope with the load of a large number of mobile terminals
7) a mechanism to assign a terminal (MMC) to a MMS

In general the assignment of a terminal to a MMS can be done either in a static way, or dynamically, but only when the terminal switches on, or dynamically and also allowing changes during the time when the terminal is active. More over we can consider two different scenarios for such an assignment:

- intra-domain: all the target MMSs belong to the same organization;
- inter-domain: the target MMSs belong to different organizations;

In the intra-domain case we can envisage that the main reason to perform an assignment to a different MMS could be the load sharing among the MMSs.

In the inter-domain case we can envisage that the main reason to perform an assignment to a different MMS could be to choose a MMS that is “close” to the terminal (MMC) and it is therefore able to provide a good packet level QoS for the media flows. We can assume that the MMC has one “home” MMS provided by its own organization and that it can roam among “visited” MMS. We could think at the choice of a MMS similarly to a “roaming” decision, where the terminal (MMC) should choose the best visited MMS to roam in when the connection to the visited MMS is better than the connection to its “home” MMS.

In this chapter we propose a signaling solution to allow the MMC to change the used MMS, if it discovers a better MMS in term of QoS. The objective for the Mobile Terminal (MT) is be attached to the MMS that provides the best packet level QoS (and consequently the best voice quality). For this reason, beside the signaling procedure to allow MMC to change its MMS, we will also discuss a mechanism for real-time QoS measures between MMC and the set of target MMSs in the system.

While the signaling solution could also be exploited in the intra-domain case, this solution is mostly focused on the inter-domain case. The rationale of the solution is to find the best MMS (“visited” or “home”) while roaming in arbitrary access networks.

In section 4.1 we describe the solution's architecture, the MMS change criteria and the Signaling requirement. In section 4.2 we show the new SIP signaling procedures that we are implemented for this solution, and how this procedures impact the canonical SIP procedures.

4.1 Overview of the proposed solution

If an MMC is allowed to change the MMS, we have to:

8) enhance the signaling specification previously showed (see section 2.4);
9) define the "criteria" that will drive the "MMS change" procedure, at least at a conceptual level.

As for the signaling, we propose to add two new signaling procedures ("MMS change" and "active probe") and to readapt the Location Update procedure.

As for the criteria, we assume that the MMC is able to know which MMSs are online and will estimate the quality of the connection towards these MMSs. To this purpose, the MMC will exchange with the MMS probe packets through each his interfaces. These packets (see section 2.5.2) are used by the MMC for monitoring the status of the connections towards the MMSs in term of jitter, packets loss and Round Trip Time (RTT). The MMC does not exchange probe packets only with the MMS that it is "attached" to, but also with each known MMS (actually with each "target" MMS).

In this solution we do not specify how the MMC knows the set of available target MMSs, but we focus on the MMS change procedure. For demonstration purposes, or in a limited environment, one can suppose that the IP addresses of target MMSs are known a priori to the MMC. In a large scale real world solution a MMS discovery procedure should be used to have this information.

4.1.1 Architecture

The architecture of the proposed solution is represented in Figure 4.1. There is no single MMS, but a set of MMSs. All the MMSs can belong to a single organization or to a set of different organizations with "roaming" agreements. For each MMC, there is a "default" or "home" MMS.

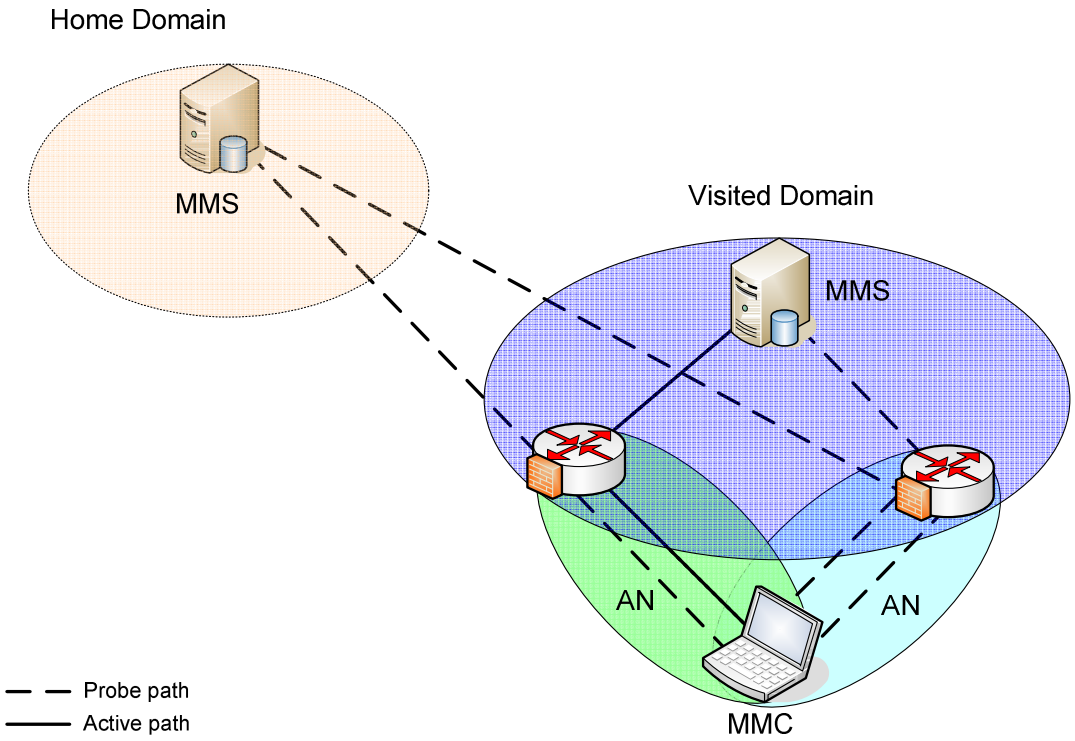


Figure 4.1: System's architecture

At a given point in time, the MMC will be handled by its home MMS or by a "visited" MMS, we will denote as "current MMS" the MMS that is handling an MMC. An MMS change procedure will change the "current MMS", in particular we will use the notation of "old MMS" (oMMS) and "new MMS" (nMMS) to indicate the MMS in charge of the MMC respectively before and after the change MMS procedure.

As we will see in more detail in the next section, the MMS change procedure will happen in phases, during a first phase the oMMS is still involved in the signaling path acting as a SIP proxy from the nMMS to the user's SIP "Home Server" and vice versa. The set of candidates MMS which the MMC may choose as "new MMS" at a given point in time / place on the earth is called the set of "target MMSs".

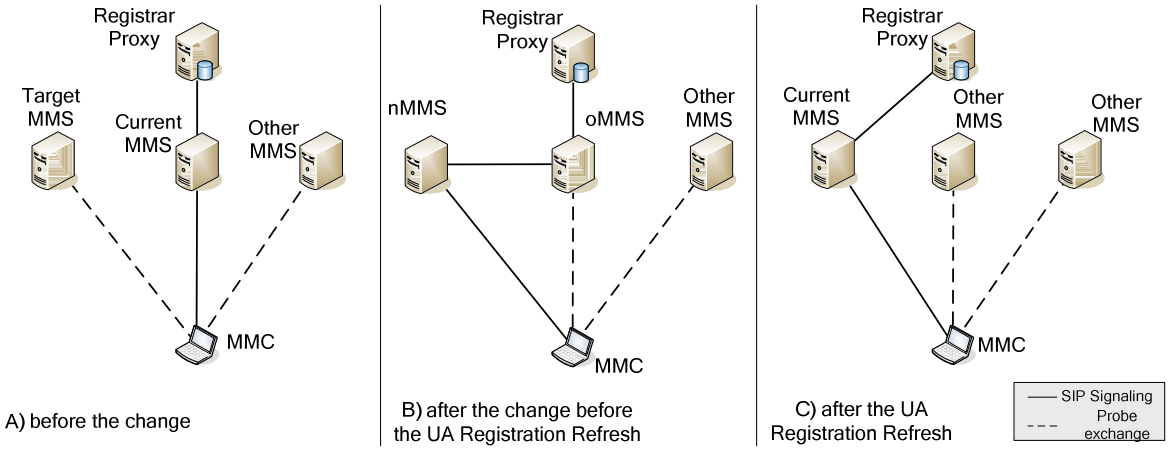


Figure 4.2: The phases of MMS change procedure

In phase B the phone is able to make and receive calls. In this case the oMMS acts as SIP signalling proxy and the nMMS acts as mobility management server and media proxy. Note that (Figure 4.3) the oMMS is not involved in media path.

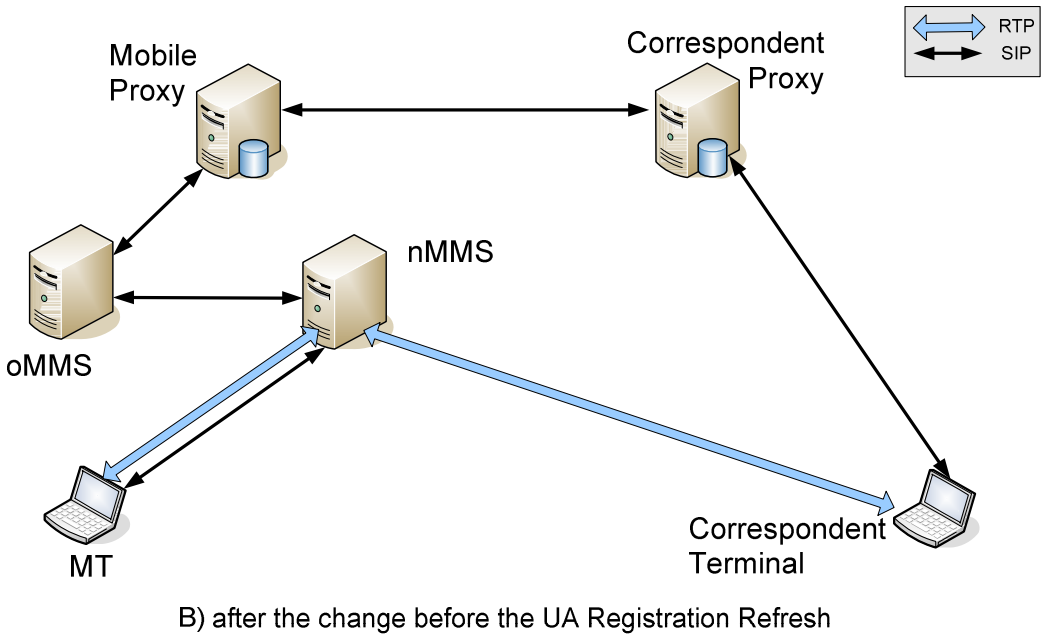


Figure 4.3: Sip Signaling and RTP flows paths in the 2nd phase to MMS change procedure

The MMS change procedure will not be performed during an active call. In fact, according to the proposed approach also the media flows would have to pass through the oMMS when the MMC is on-call. In this manner, we would not improve the quality of the media flow. Therefore, the only possible handover during an active call is the one between different

wireless interfaces of a mobile terminal, but the mobile terminal will communicate with the current MMS during the whole lifetime of a call.

4.1.2 MMS change Criteria

The decision to change MMS is taken by the MMC, taking into account the status of the connections to each target MMS. If the connection towards a target MMS has better QoS parameters (jitter, RTT and packets loss) than those of the current MMS, the MMC can activate the "MMS change procedure" to attach itself to anther MMS.

Note that the MMC needs to correlate the different measurements of the connections towards the MMSs over each active interface. For example, if only one interface is getting worse, the MMS change procedure should not be used because, in this case, likely the access network is the bottleneck. So the MMC can change the active interface with a "normal" handover (location update procedure).

4.1.3 Signaling requirement

In order to introduce the new procedures (MMS change and active probe) we have to modify the SIP REEGISTER message that has been called "Location Update" Register in section 2.4.1. In fact this message now will be used to:

- indicate the wireless interface in use to the current MMS;
- change the current MMS;
- activate the packet probe exchange with a target MMS;
- change the MMS role.

Therefore we need to add two new parameters that will be include in the Contact Header to Location Update Register. This parameters are named "MMS-role" and "oldMMS".

The "MMS-role" parameter is used to communicate the MMS which is the role that needs to be played by the MMS. The "oldMMS parameter" shows which MMS the MMC was previously attached to.

The MMC take a decision about the role of the MMS, which can have three roles:

- **Mobility Manager:** the MMC and the MMS manage the terminal mobility. In this case the value is: "*MMS-role=currentMMS*". If the MMS is already attached to another MMS, it adds the parameter "*oldMMS=ip_address*" to communicate to the new MMS which is the previous MMS used.

- **Prober:** the MMC and the MMS can exchange only the probe packets for monitoring to QoS. In this case the value of MMS-role parameter is *proberMMS*.
- **Proxy:** the MMS that has received a message with "*MMS-role=proxyMMS*" must work as a "canonical" SIP proxy for the terminal-ID in the contact header. This message is sent from an MMS only if it has received a LU with a parameter *oldMMS* to the IP address in the oldMMS value.

4.2 Signaling Procedures

4.2.1 Location Update

The basic LU procedure is not changed moving from the centralized solution previously described to this solution with a multiplicity of MMSs. The main purpose of the LU procedure is to allow the MMC to communicate to the MMS what is the interface in use. The LU procedure is performed when the client is off-call. However, in this proposal the LU Register message will be used not only for this purpose, but also for the "Active Probe Setup" and for the "MMS change" procedures, we need to add the "MMS-role" parameter with value "*currentMMS*" to the Contact header:

```
REGISTER sip:160.80.82.27:5070 SIP/2.0
Via: SIP/2.0/UDP
83.225.138.116;MMID=user@iptel.uniroma2.it;branch=z9hG4bKd7b
d1
Max-Forwards: 70
To: <sip: user@iptel.uniroma2.it >
From: <sip: user@iptel.uniroma2.it >;tag=4758d7f7
Call-ID: 4614a25233b6f9f5@user
CSeq: 1 REGISTER
Contact: <sip:user@83.225.138.116>;MMS-role=currentMMS
Expires: 3600
Content-Length: 0
```

Note that when a MMC completes the Location Update procedure it will be logically attached to MMS. The MMS becomes its anchor point managing its mobility.

With respect to the solution proposed in chapter 2, we now foresee that all the links between the MMC and the MMS (i.e. all the available wireless interfaces on the MMC) can be monitored by the periodic exchange of probe packets.

4.2.1.1 Active Probe

We assume that the MMC can monitor the quality of the links connecting towards the different "target" MMS, in order to choose the most suitable MMS.

As described in 2.5.2 they are two kinds of link's quality monitoring:

- **active monitoring:** this procedure requires to inject probe traffic in the link;
- **passive monitoring:** this procedure is used to monitor the existing traffic in the link without inserting any new packet;

In our case we need to use an "active" procedure because we want to monitor the link quality also when no media packet transit in this link. The procedure shown in section 2.5.2 matches this requirement.

The MMC send a LU register with "MMS-role=proberMMS" to each known MMSs. This message is used to start up the probe packets exchange. Through this exchange the MMC can monitor the QoS of the links between MMC and MMS. This message is sent to each known target MMSs.

```
REGISTER sip:160.80.82.27:5070 SIP/2.0
Via: SIP/2.0/UDP
83.225.138.116;MMID=user@iptel.uniroma2.it;branch=z9hG4bKd7
bd1
Max-Forwards: 70
To: <sip: user@iptel.uniroma2.it >
From: <sip: user@iptel.uniroma2.it >;tag=4758d7f7
Call-ID: 4614a25233b6f9f5@user
CSeq: 1 REGISTER
Contact: <sip:user@83.225.138.116>;MMS-role=proberMMS
Expires: 3600
Content-Length: 0
```

In this case the MMC does not use the mobility features of the MMS because it is attached to a different MMS. However, if the quality parameters are better than the current MMS the MMC can activate the MMS change procedure.

The idea of making active measurements to monitor the performance of a connection when the actual media is not active is also discussed within the IETF mmusic WG. A WG internet draft [1] proposes the SDP loop back approach. This is an SDP extension that allows to establish a loop back connection for monitoring the performance of the media transport.

[1] draft-ietf-mmusic-media-loopback-06

4.2.2 MMS change

This procedure is used by the MMS in order to change the current MMS when the quality of each link gets worse. Figure 4.4 shows the message flows:

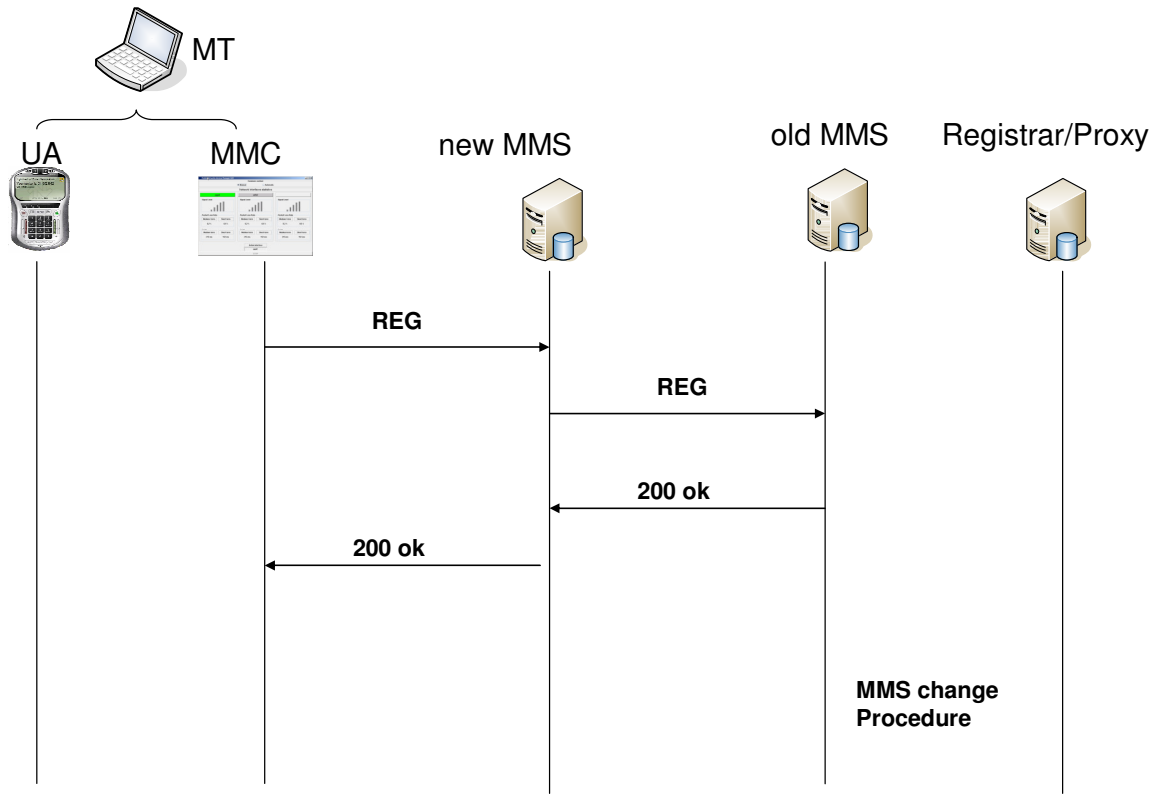


Figure 4.4: MMS change procedure

When a MMC wants to change its current MMS (oMMS), it will send an LU Register to another MMS (nMMS) with parameter *"MMS-role=currentMMS"*. Moreover it will set the parameter *oldMMS* with the main[2] oMMS's IP address.

[2] Each have more IP addresses

```
REGISTER sip:160.80.82.27:5070 SIP/2.0
Via: SIP/2.0/UDP
83.225.138.116;MMID=user@iptel.uniroma2.it;branch=z9hG4bKd7bd1
Max-Forwards: 70
To: <sip: user@iptel.uniroma2.it >
From: <sip: user@iptel.uniroma2.it >;tag=4758d7f7
Call-ID: 4614a25233b6f9f5@user
CSeq: 1 REGISTER
Contact: <sip:user@83.225.138.116>; MMS-
role=currentMMS;oldMMS=160.80.82.12
Expires: 3600
Content-Length: 0
```

After than nMMS received this message, it send a LU to oMMS with parameter "*MMS-role=proxyMMS*".

```
REGISTER sip:160.80.82.27:5070 SIP/2.0
Via: SIP/2.0/UDP
83.225.138.116;MMID=user@iptel.uniroma2.it;branch=z9hG4bKd7bd1
Max-Forwards: 70
To: <sip: user@iptel.uniroma2.it >
From: <sip: user@iptel.uniroma2.it >;tag=4758d7f7
Call-ID: 4614a25233b6f9f5@user
CSeq: 1 REGISTER
Contact: <sip:user@83.225.138.116>; MMS-role=proxyMMS
Expires: 3600
Content-Length: 0
```

In this way the oMMS stop to act as the mobility manager for the MMC and acts as canonical proxy SIP for it. The nMMS set the oMMS as outbound proxy for this MMC, so it forwards each request sent from MMC to oMMS.

This is needed because the UA (into MT) is registered to Registrar server with the oMMS IP address, so if the oMMS is not in the path, the UA is unreachable.

We enter in a so called "transition phase", until a new User Registration will be sent by the user to the User Registrar server, which will replace the oMMS with the nMMS in the main Registrar Server.

4.2.2.1 Incoming and Outgoing call

During the "transition phase" the SIP requests will be sent to the old MMS and it will forward these messages to new MMS that knows which is the MT current IP address.

The call flows for both kinds of calls (outgoing/incoming) are modified accordingly. In fact, in the incoming call scenario the MT Proxy forwards the INVITE to oMMS, the oMMS acts as proxy and it forwards the message to nMMS.

It is the nMMS that modifies the message to take cero of the user mobility. Therefore the RTP flows will only need to cross the media proxy into the new MMS.

Figure 4.5 shows this scenario.

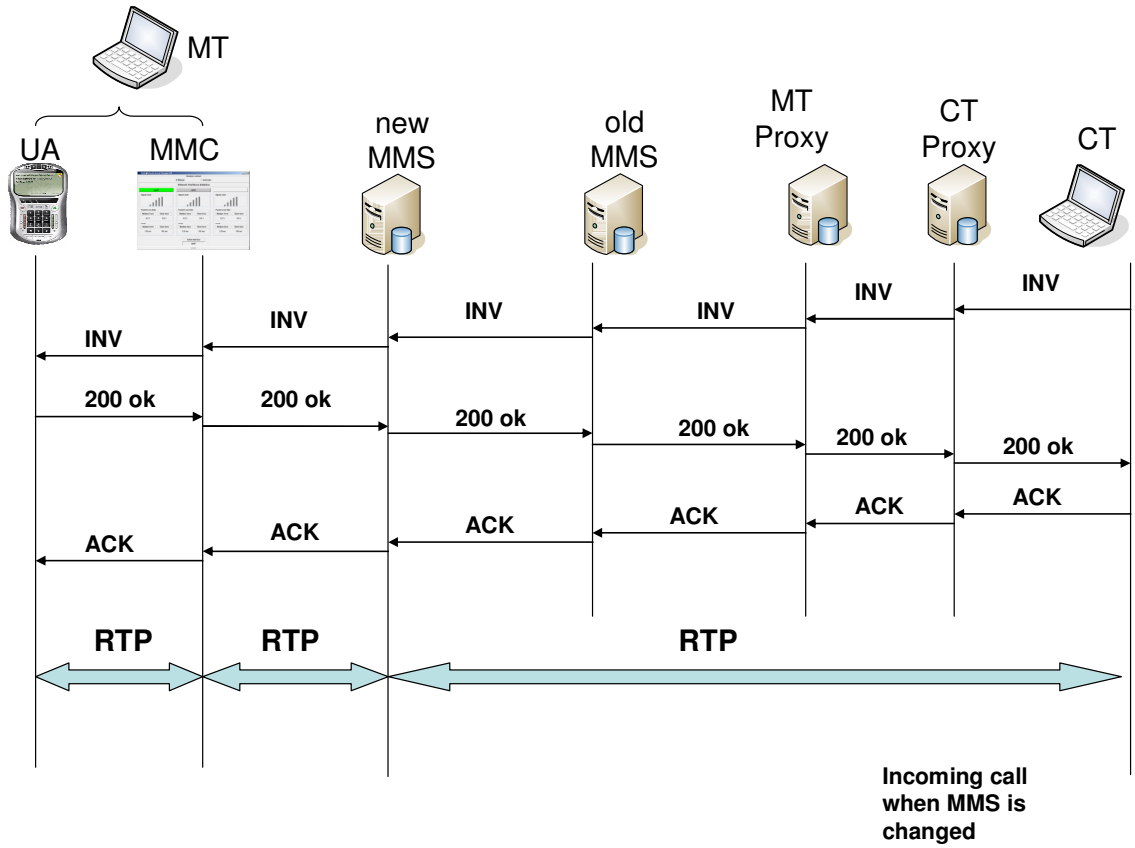


Figure 4.5: Incoming call after the 2nd phase to MMS change procedure

Also in outgoing call scenario, when the nMMS receives an Invite request from MT, it modifies the message to take care of the MT mobility and forwards this message to the oMMS. The oMMS will not do any modification to message, it only will forward it to the MT outgoing proxy (as Figure 4.6) or directly to the corresponding CT proxy if the MT outgoing proxy is not needed.

Note that both in incoming and in outgoing calls the media flows do not transit through the oMMS but only through the nMMS. The oMMS is only involved in the signaling path.

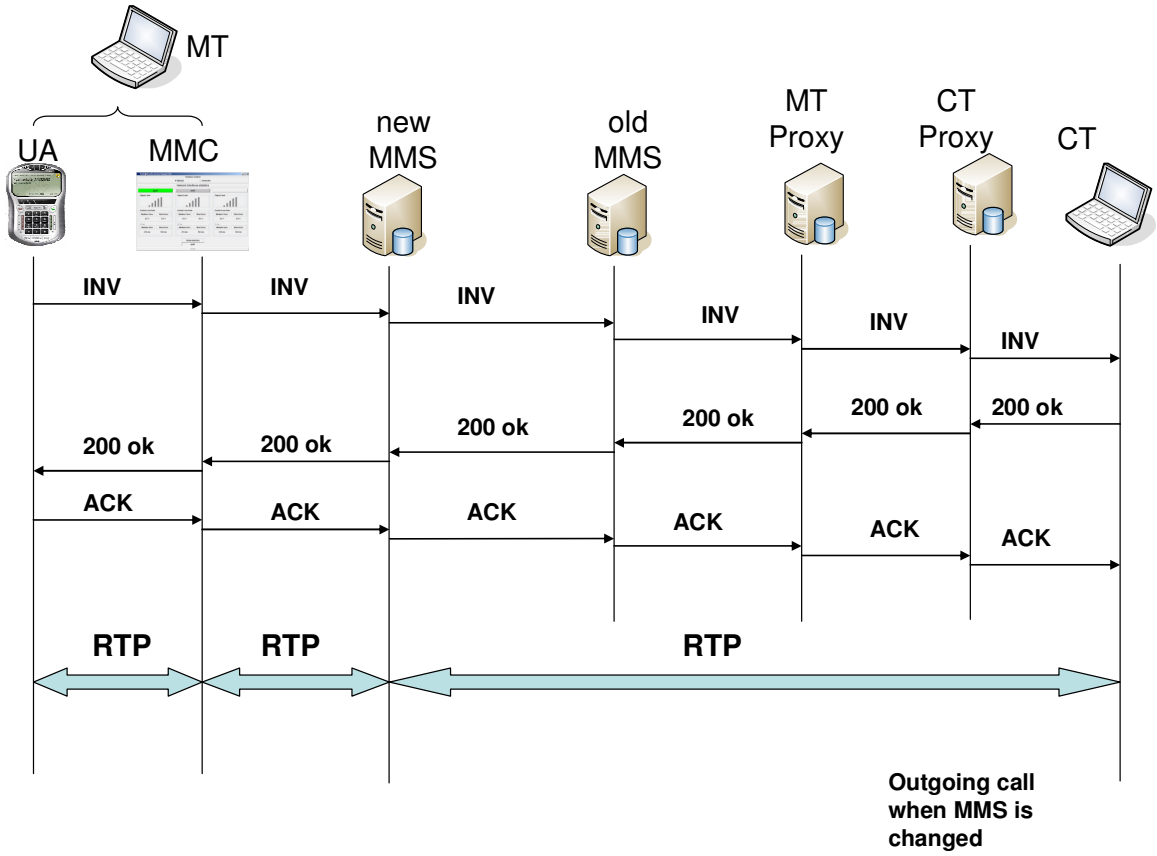


Figure 4.6: Outgoing call after the 2nd phase to MMS change procedure

4.2.2.2 User Registration Refresh

The old MMS will remain in the MT call flows during the transition phase, i.e. until when the UA into MT will refresh his user registration.

In fact, when the new MMS receives a user registration refresh, it changes the contact header with its own IP address. In this manner when the Registrar proxy receives this message, it removes the old MMS address and set the nMMS IP address as MT contact address. Hence each request for the MT will be forwarded directly to the nMMS.

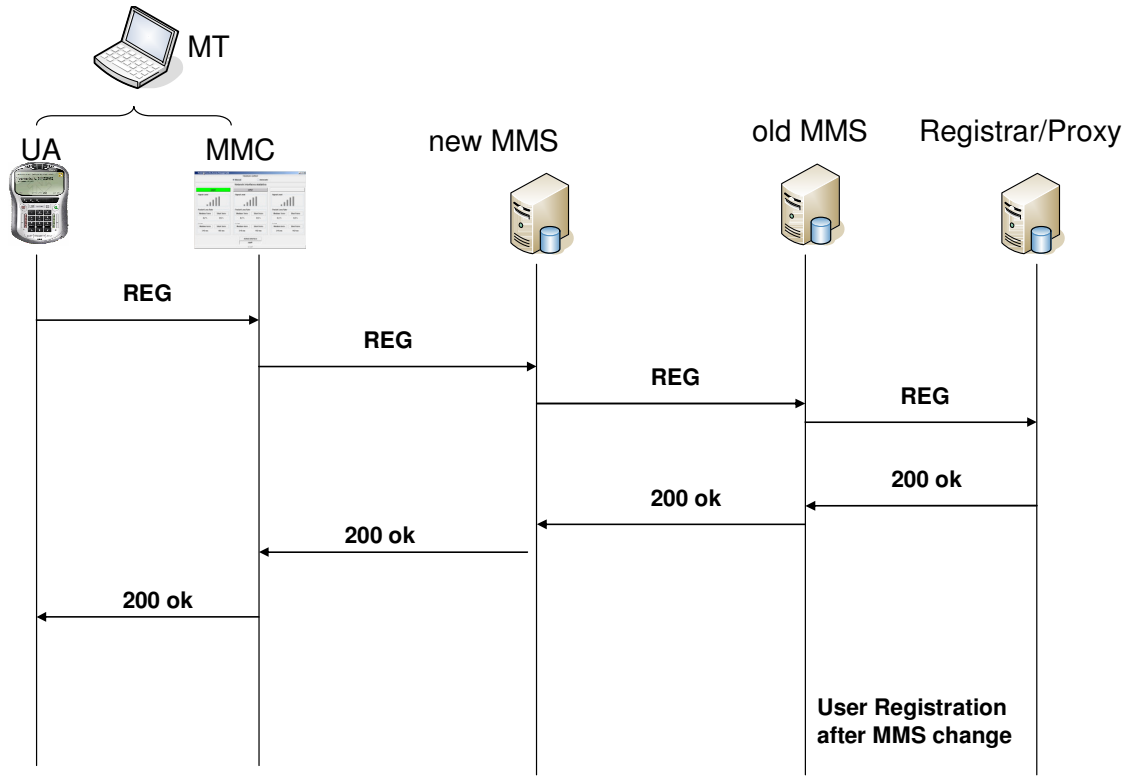


Figure 4.7: User registration refresh after the 2nd phase to MMS change procedure

The most important assumption here is that the expire time of the registration is shorter than the expire time recommended in SIP RFC 3261, so that the oMMS will be involved in the signaling path as shortly as possible.

In order not to insert the oMMS in the signaling path, we can use a forced re-registration procedure as it is described in RFC 3680. .This procedure allows the registrar server to require a new registration to a user agent that is subscribed at this service. In this case, the new nMMS should sent a notify to UA to force a new registration. Note that it is not completely correct in SIP, because this message should not be sent by the nMMS, by Registrar Server. Moreover the Registrar server is seamless at the change MMS procedure.

Conclusions

In this thesis we have presented MMUSE, a solution for seamless vertical handover between heterogeneous networks like WiFi and 3G, based on SIP.

The MMUSE key idea is to improve the capabilities of a SIP "Session Border Controller" (SBC), in order to support the SIP client mobility. The proposed solution can be exploited in the short term by a 3G operator willing to extend its services to WLAN, by a VoIP provider that uses the 3G network as IP transport and by an enterprise that wants to directly manage its voice services. In the long term this kind of approach will likely need to be included in NGN networks, which aim to support communication over heterogeneous networks (including legacy networks) in a seamless way. All the proposed mechanisms have been implemented within a test-bed that fully demonstrates the correctness and simplicity of the solution. Moreover, significant measurement tests have also be run in order to provide quantitative evaluation of the roaming solution.

Currently the requirements for a mobility management solution and the signaling procedures shown in this thesis have been proposed to IETF Sipping Working Group in 2 drafts [4] and [5].

In order to compare MMUSE with other handover solutions in literature we have proposed a model to classify the mobility management solutions for vertical handover. The purpose of the model is to compare the handover solutions in term of disruption time both in the ideal case (without packet loss) and taking into account packet loss. The model has been used to compare the two reference solutions for mobility management at network lever (MIPv4) and application lever (SIP-reINVITE) with MMUSE.

The analysis has shown that the performances of MIPv4 and SIP-reINVITE mainly depend on the delay between the Mobile Node and the Correspondent Node, while the performance of MMUSE mainly depends of the delay between the Mobile node and the MMS node. If the location of the MMS can be controlled, MMUSE can easily outperform the other solutions.

The analysis confirmed that MMUSE performs better than other solutions also in presence of failures and retransmissions.

The most critical weakness of version of MMUSE that we have implemented is the scalability. In the MMUSE architecture defined in chapter 2 the MMS could be a bottleneck because it is a proxy for both media and signaling packet, for each MMC. In order to improve

MMUSE scalability we defined a solution that foresees a set of cooperating MMS and allows the MMC to dynamically change the MMS based on QoS parameters criteria.

Future works include the possibility to extend MMUSE to become a mobility management solution not only for SIP applications but for any kind of application. A possible MMUSE improvement in this direction in under study in the research project PERIMETER, funded by the EU under the 7-th Framework Programme.

References

[1] S. Salsano, L. Veltri, G. Martiniello, A. Polidoro, "Seamless vertical handover of VoIP calls based on SIP Session Border Controllers" IEEE International Conference on Communications, ICC'06, 11-15 June 2006, Istanbul, Turkey

[2] S. Salsano, L. Veltri, A. Polidoro, A. Ordine "Architecture and testbed implementation of vertical handovers based on SIP Session Border Controllers", Wireless Personal Communications, Springer, published on-line March 31, 2007, doi: 10.1007/s11277-007-9259-2

[3] Stefano Salsano, Chiara Mingardi, Saverio Niccolini, Andrea Polidoro, Luca Veltri,SIP-based Mobility Management in 4G networks, IEEE Wireless Communication Magazine Vol. 15, Issue 2, April 2008, Page(s): 92-99, doi:10.1109/MWC.2008.4492982

[4] S. Niccolini, S. Salsano, L. Veltri "Requirements for vertical handover of multimedia sessions using SIP", draft-niccolini-sipping-siphandover-05, Work in progress, May 2009

[5] S. Salsano, S. Niccolini, L. Veltri, A. Polidoro "A solution for vertical handover of multimedia sessions using SIP" draft-salsano-sipping-siphandover-solution-02, Work in progress, November 2008

[6] Andrea Polidoro, Saverio Niccolini, Stefano Salsano, Performance evaluation of vertical handover mechanisms in IP networks, IEEE Wireless Communication Networks Conference WCNC'08 , 31 March-3 April Las Vegas, Nevada USA.

[7] M.Fiorani, A. Labella, A. Ordine, A. Polidoro, S. Salsano, L. Veltri, "Report of architecture and measurements for SBC based vertical handovers", May 2006, available at: http://netgroup.uniroma2.it/Stefano_Salsano/SIP-SBC-seam-HO/report-2006-05.pdf

[8] J. Rosenberg et al. "SIP: Session Initiation Protocol", RFC 3261, June 2002

[9] C. Perkins, "IP Mobility Support for IPv4", RFC 3344, August 2002

[10] IEEE P802.21/D11.0 Draft Standard for Local and Metropolitan Area Networks: Media Independent Handover Services, http://www.ieee802.org/21/doctree/Temp/P802-21-D11.pdf

[11] M. Riegel et al. "Mobile SCTP" draft-riegel-tuexen-mobile-sctp-09, November 5, 2007

[12] A. T. Campbell, J. Gomez S. KIM, A. G. Valko Z. R. Turanyi, C.-Y. Wan, "Design, Implementation, and Evaluation of Cellular IP", IEEE Personal Communications, August 2000

[13] H. Schulzrinne and E. Wedlung, "Application-layer mobility using SIP", Mobile Comp. and Commun. Rev., vol. 4, no. 3, July 2000

[14] N. Banerjee, A. Acharya, S.K. Das, "Seamless SIP-Based Mobility for Multimedia Applications", IEEE Network, March/April 2006

[15] R. Mahy and D.Petrie, “The Session Initiation Protocol (SIP) Join Header”, RFC 3911, October 2004

[16] Ashutosh Dutta, Sunil Madhani, Wai Chen, Onur Altintas, Henning Schulzrinne “Fast-handoff Schemes for Application Layer Mobility Management”, PIMRC 2004, Spain.

[17] A.Dutta, B.Kim, T.Zhang, S.Baba, K.Taniuchi, Y.Ohba, H.Schulzrinne “Experimental Analysis of Multi Interface Mobility Management with SIP and MIP”, IEEE Wirelesscom 2005.

[18] T.T. Kwon, M. Gerla, S. Das, “Mobility management for VoIP service: Mobile IP vs. SIP” IEEE Wireless Communications, 10.1109/MWC.2002.1043856 (2002)

[19] H. Fathi, R. Prasad, S. Chakraborty, “Mobility management for VoIP in 3G systems: evaluation of low-latency handoff schemes” IEEE Wireless Communications, 10.1109/MWC.2005.1421933 (2005)

[20] J. Hautakorpi, Ed. “Requirements from SIP (Session Initiation Protocol) Session Border Control Deployments”, draft-ietf-sipping-sbc-funcs-04.txt, December 2007

[21] J. Cumming “Sip Market Overview” http://www.dataconnection.com/ network/download/whitepapers/sipoverview.pdf

[22] J. Rosenberg, J. Weinberger, C. Huitema, R. Mahy, "STUN - Simple Traversal of User Datagram Protocol (UDP) Through Network Address Translators (NATs)", IETF RFC 3489, March 2003

[23] Rosenberg, J., "Obtaining Relay Addresses from Simple Traversal Underneath NAT (STUN)", draft-ietf-behave-turn-02 (work in progress), October 2006.

[24] MediaProxy, http://mediaproxy.ag-projects.com/README

[25] M. Handley and V. Jacobson, “SDP: Session Description Protocol,” IETF RFC 2327, April 1998

[26] J. Rosenberg and H.Schulzrinne, “An Extension to the Session Initiation Protocol (SIP) for Symmetric Response Routing”, IETF RFC 3581, August 2003..

[27] R.G. Cole, J.H. Rosenbluth, “Voice over IP Performance monitoring”, ACM SIGCOMM Computer Communication Review, Volume 31 , Issue 2, Pages: 9–24 (April 2001)

[28] ITU-T Recommendation G.107, “The E-Model, a computational model for use in transmission planning”, December 1998.

[29] N. Banerjee, Wei Wu; S.K. Das, “Mobility support in Wireless Internet”, IEEE Wireless Communications, vol 10, no 5, 2003, pp54-61

[30] M. M. Buddhikot, G. Chandranmenon, S. Han, Y.-W. Lee, S. Miller, L. Salgarelli, “Design and Implementation of a WLAN/CDMA 2000 Interworking Architecture”, IEEE Communication Magazine, November 2003.

[31] Giuseppe Ruggeri, Antonio Iera, Sergio Polito: “802.11-Based Wireless-LAN and UMTS interworking: requirements, proposed solutions and open issue”. Computer Networks 47(2): 151-166 (2005).

[32] Xlite SIP User Agent, http://www.counterpath.com/

[33] MjSip open source Java SIP stack, http://www.mjsip.org

[34] SER free SIP server, http://www.iptel.org

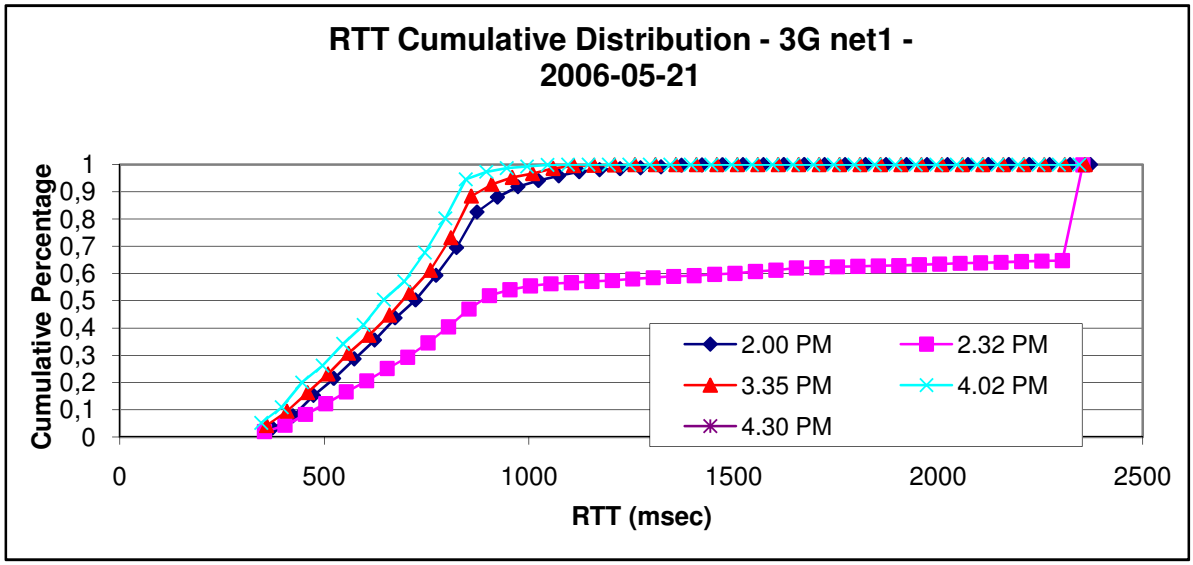
[35] G. Combs *et al.*, "Ethereal: A Network Protocol Analyzer", http://www.ethereal.com/, now wireshark http://www.wireshark.org/

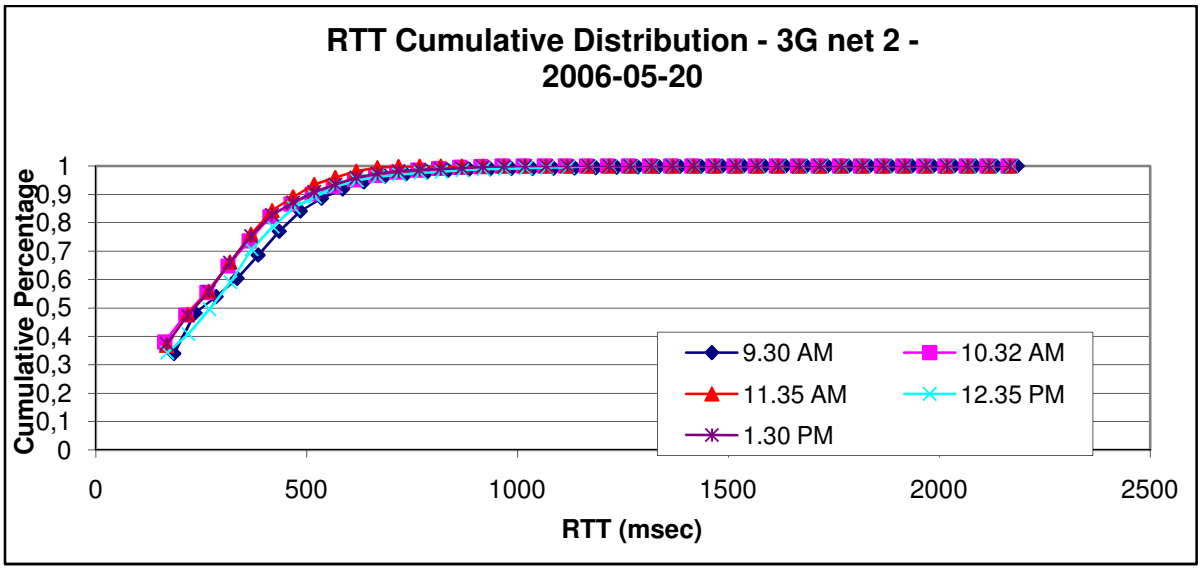
[36] BlueZ, Official Linux Bluetooth protocol stack, http://www.bluez.org/

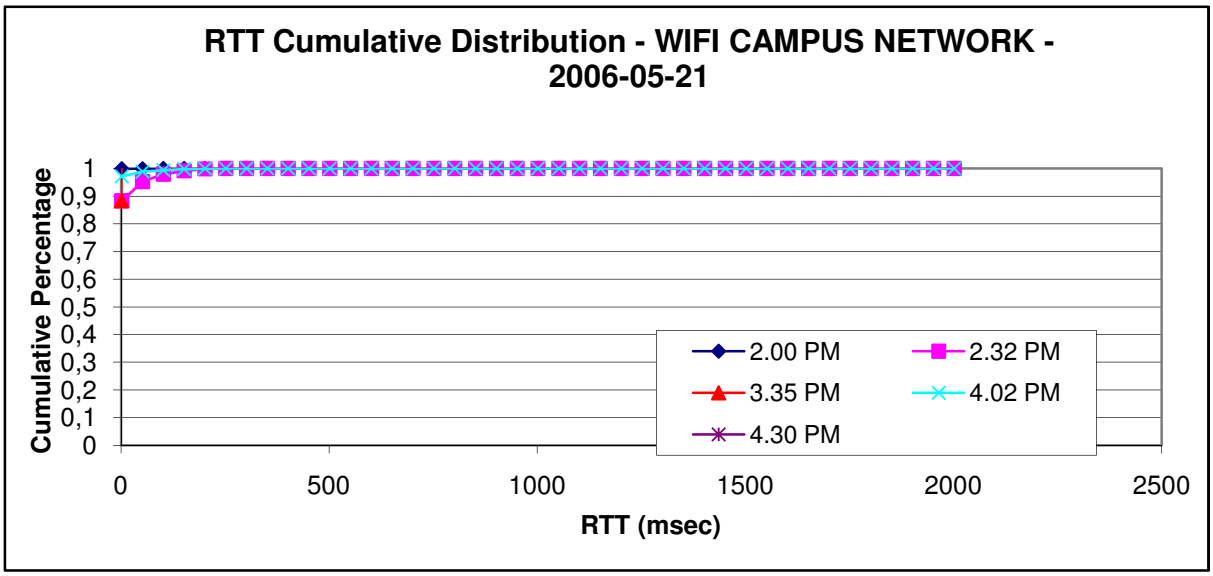
[37] PERIMETER European Project: http://www.ict-perimeter.eu/

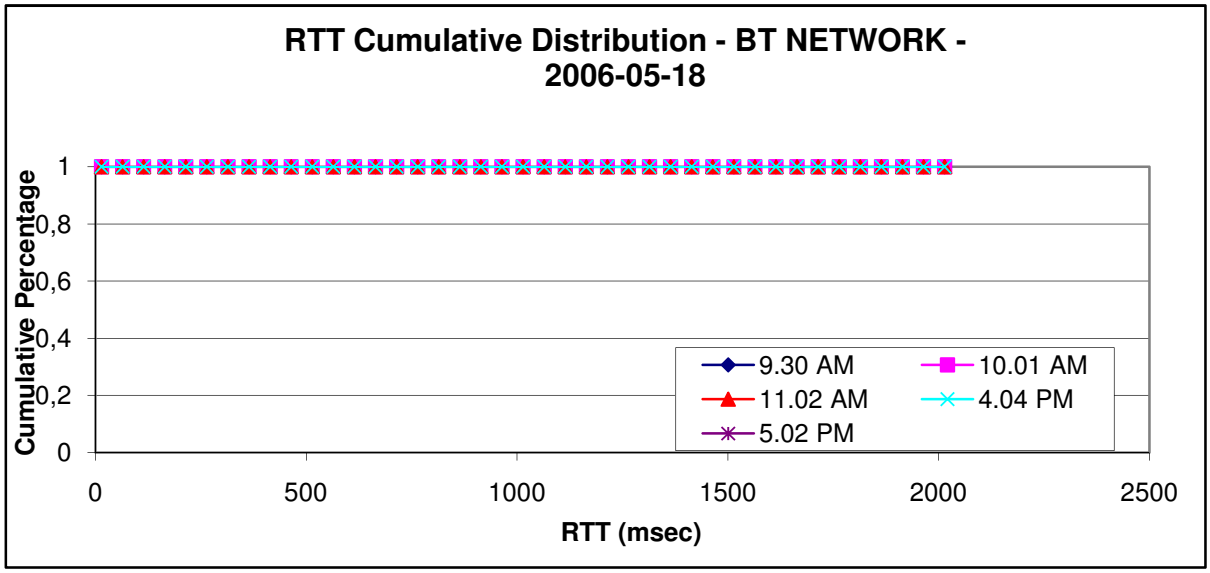
Appendix A: Report of MMUSE measurements

1 Access network performance evaluation





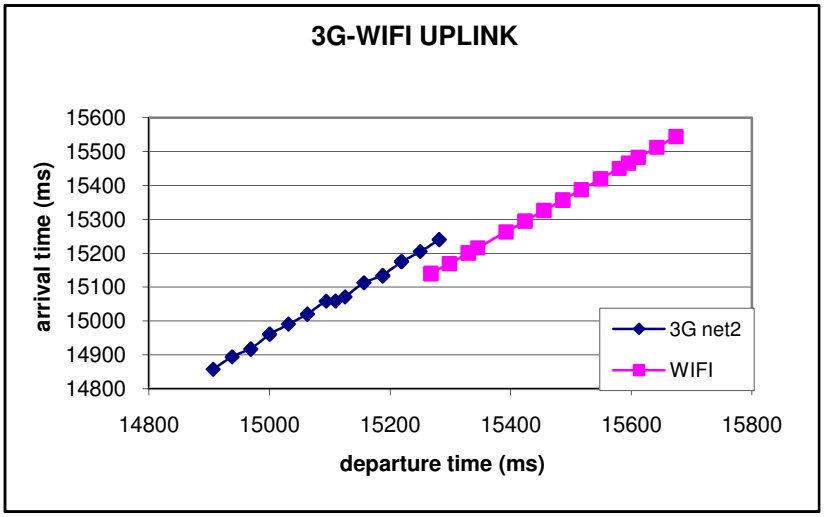

RTT Cumulative Distribution - WIFI CAMPUS NETWORK - 2006-05-21
Cumulative Percentage
1
0,9
0,8
0,7
0,6
0,5
0,4
0,3
0,2
0,1
0
0
500
1000
1500
2000
2500
RTT (msec)
2.00 PM
2.32 PM
3.35 PM
4.02 PM
4.30 PM

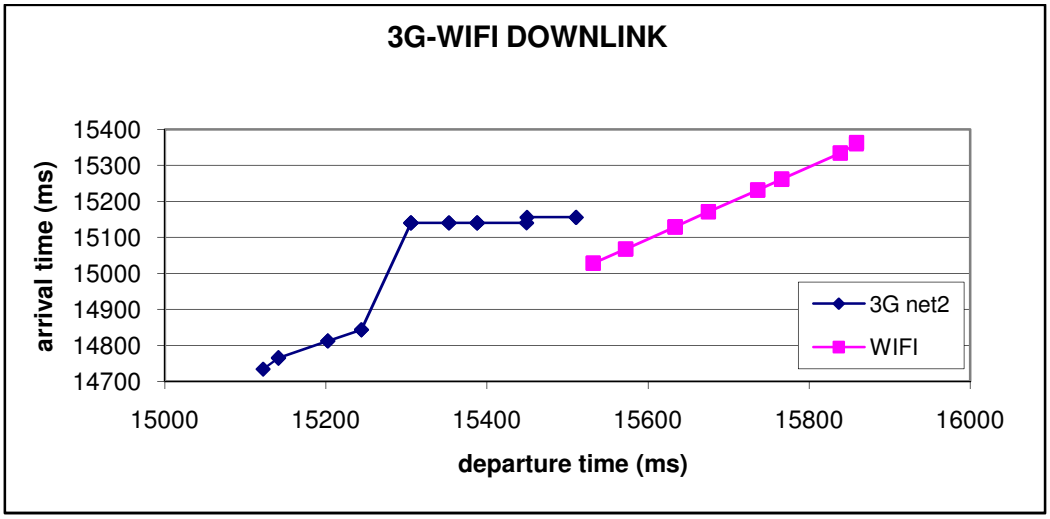

RTT Cumulative Distribution - BT NETWORK - 2006-05-18
Cumulative Percentage
1
0,8
0,6
0,4
0,2
0
0
500
1000
1500
2000
2500
RTT (msec)
9.30 AM
10.01 AM
11.02 AM
4.04 PM
5.02 PM

2 Evaluation of handover performance

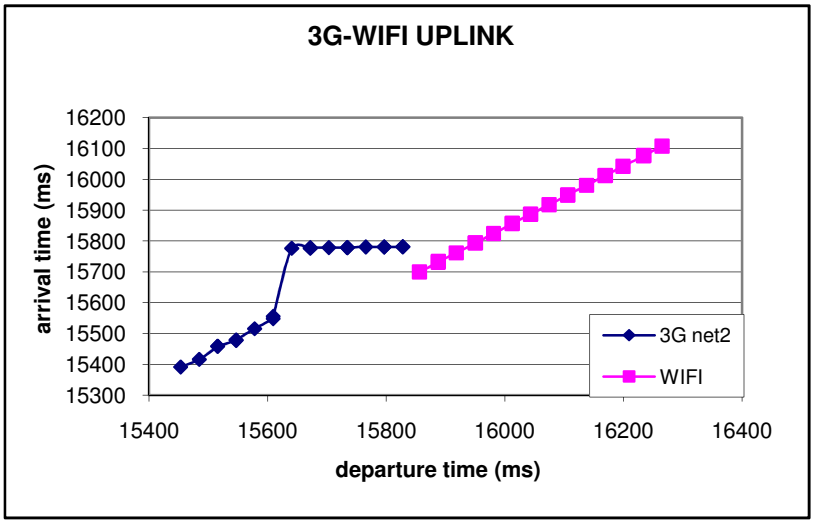
2.1 HANDOVER 3G (net2) – WIFI CAMPUS NETWORK

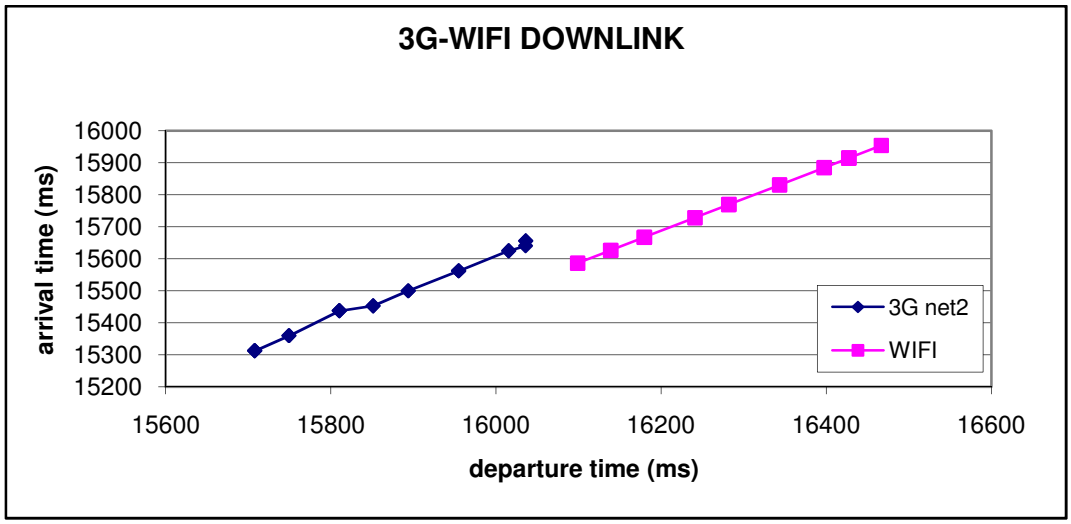
2.1.1 1^ measure: 2006-05-20, 10.32 am



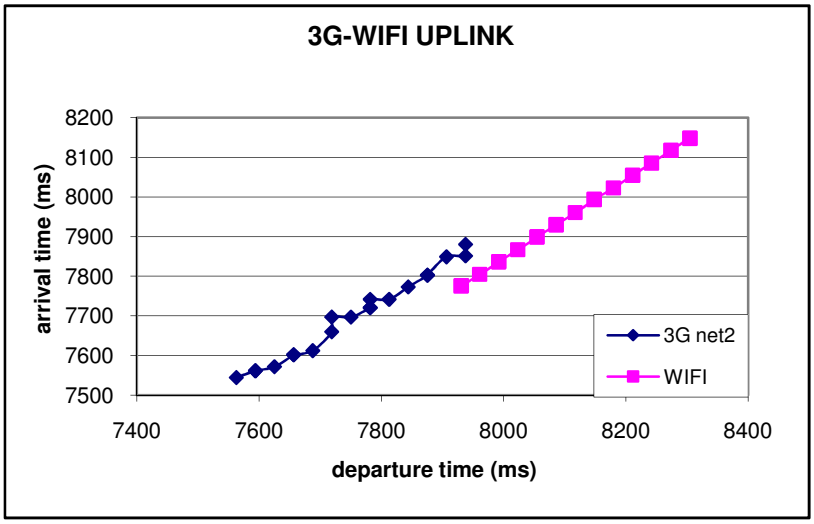


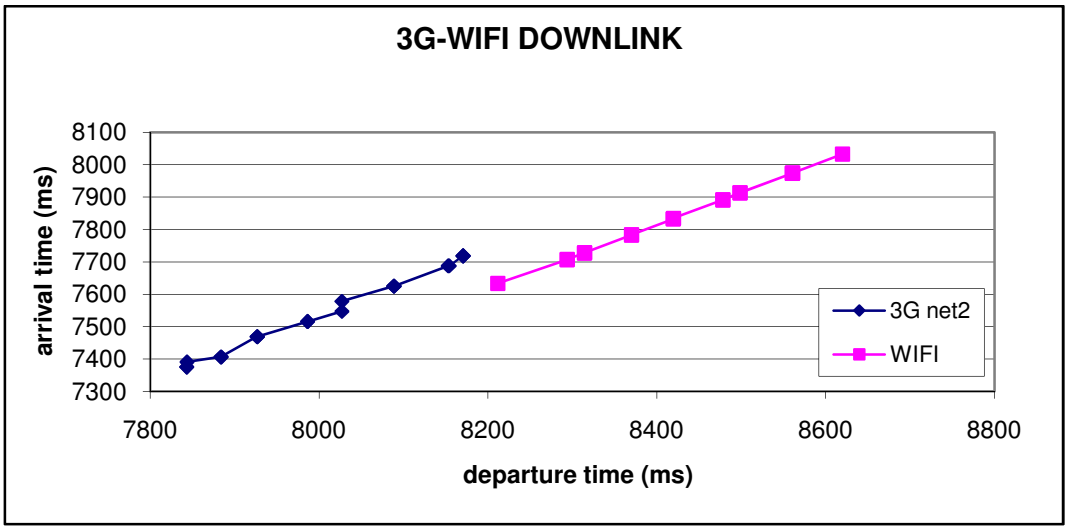
2.1.2 2^ measure: 2006-05-20, 11.35am





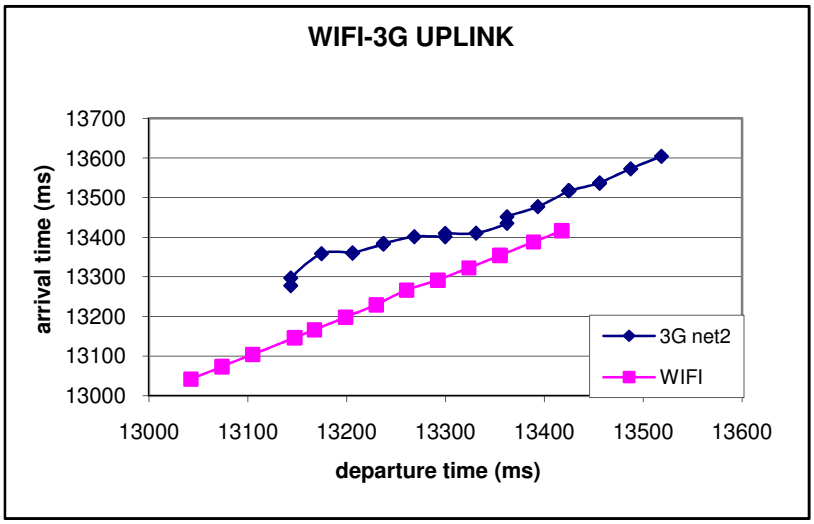
2.1.3 3^ measure: 2006-05-20, 12.35 am

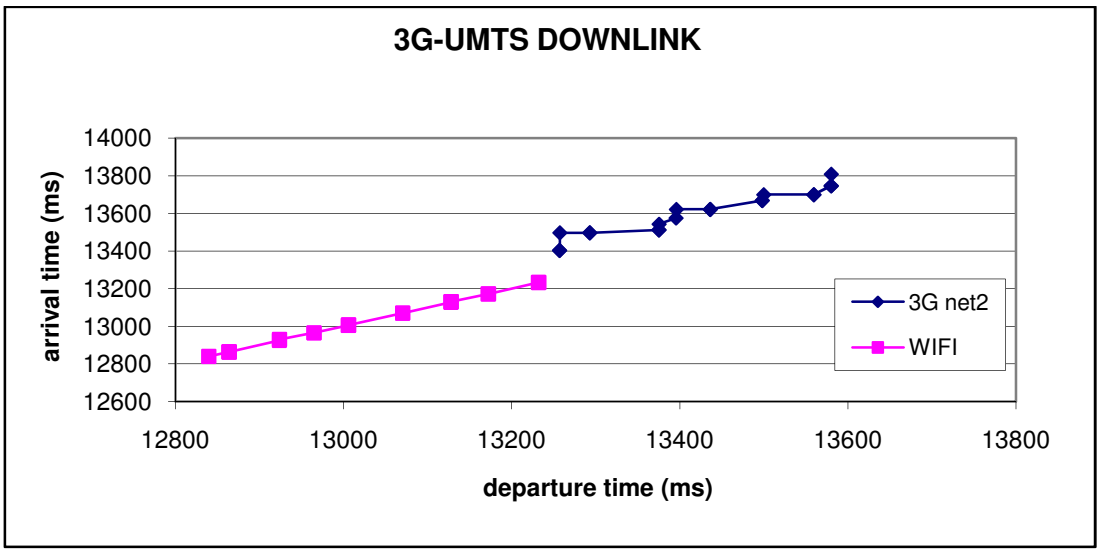




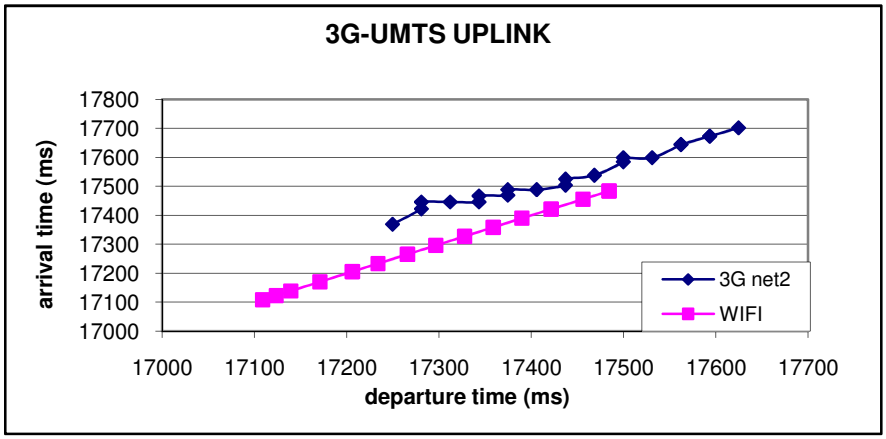
2.2 HANDOVER WIFI CAMPUS NETWORK – 3G (net2)

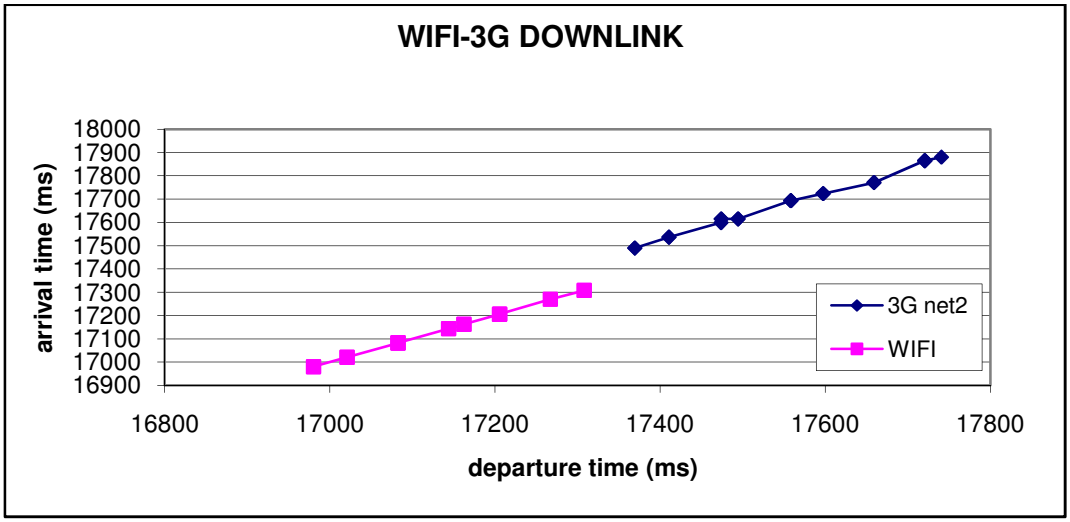
2.2.1 1^ measure: 2006-05-20, 10.45 am



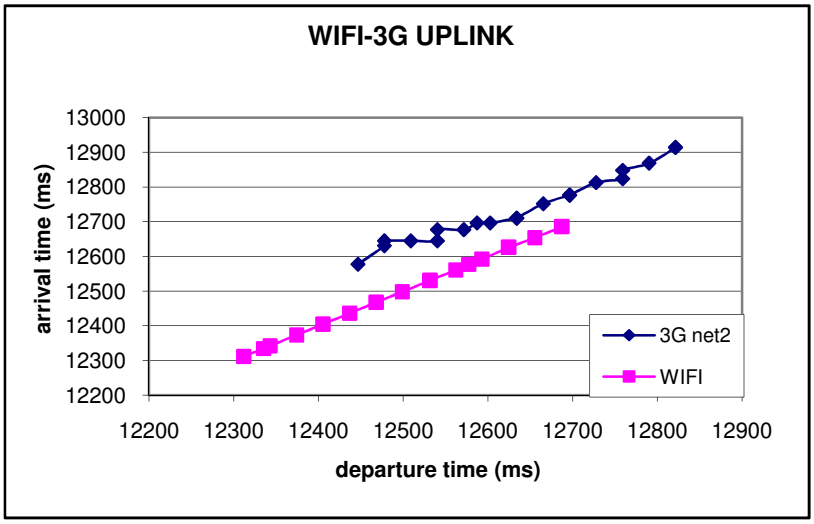


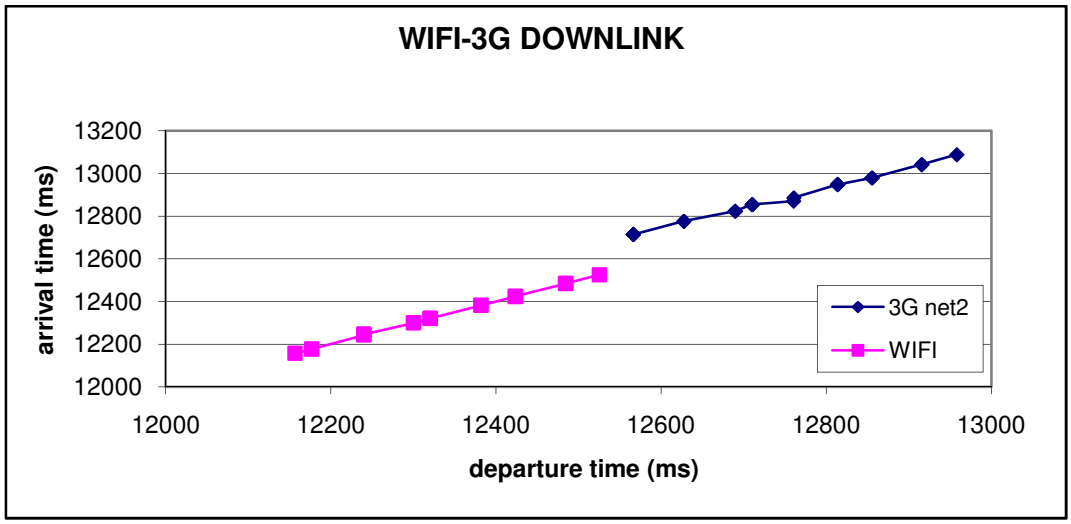
2.2.2 2^ measure: 2006-05-20, 11.47am





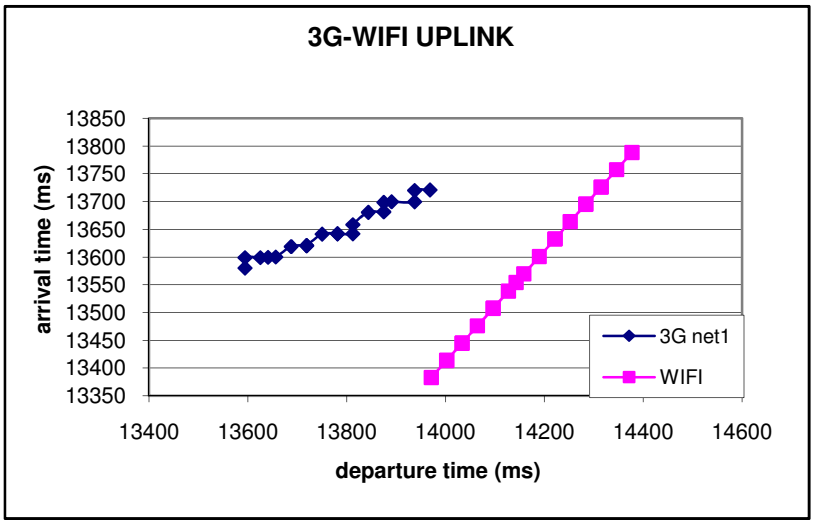
2.2.3 3^ measure: 2006-05-20, 12.50am

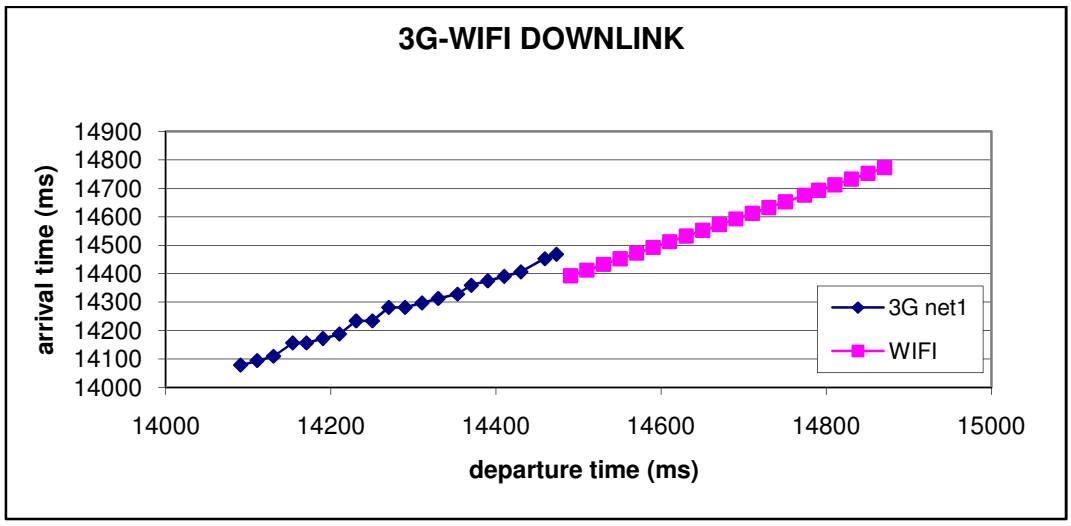




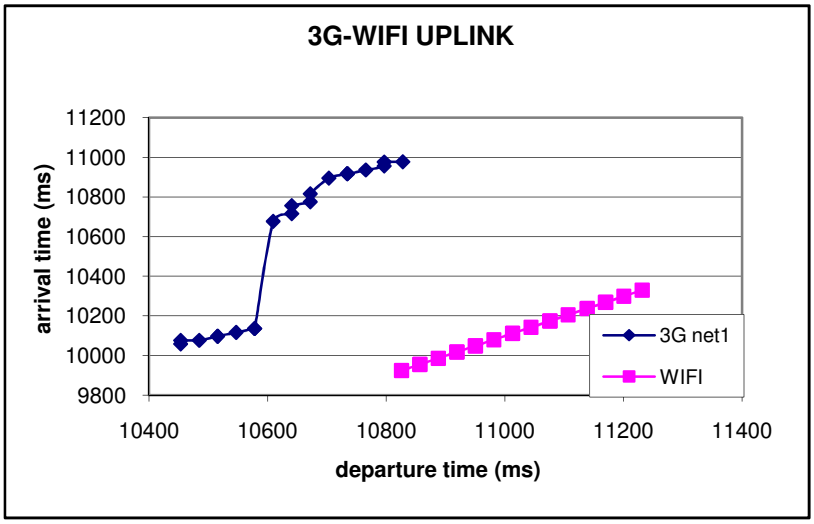
2.3 HANDOVER 3G (net 1) – WIFI CAMPUS NETWORK

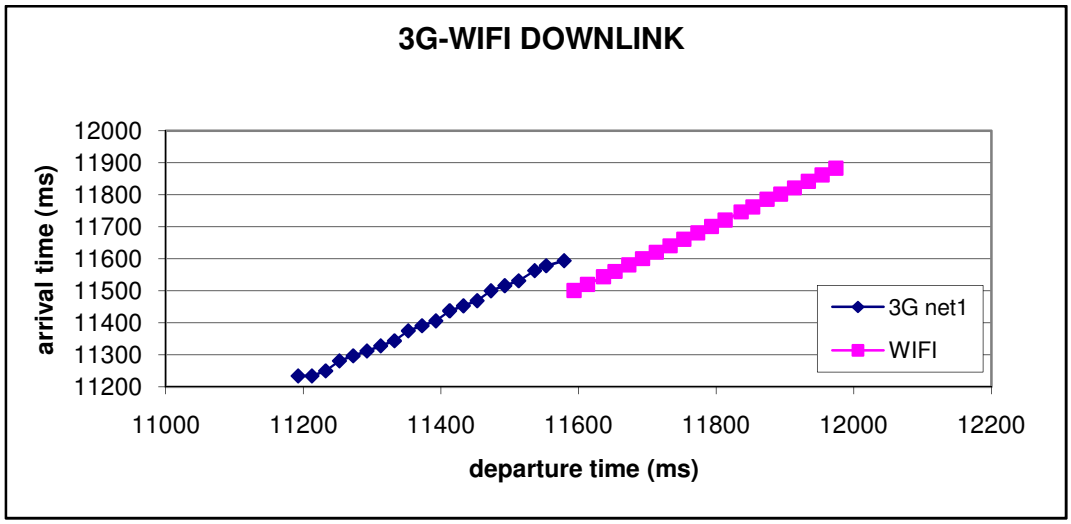
2.3.1 1^ measure: 2006-05-21, 2.02 pm





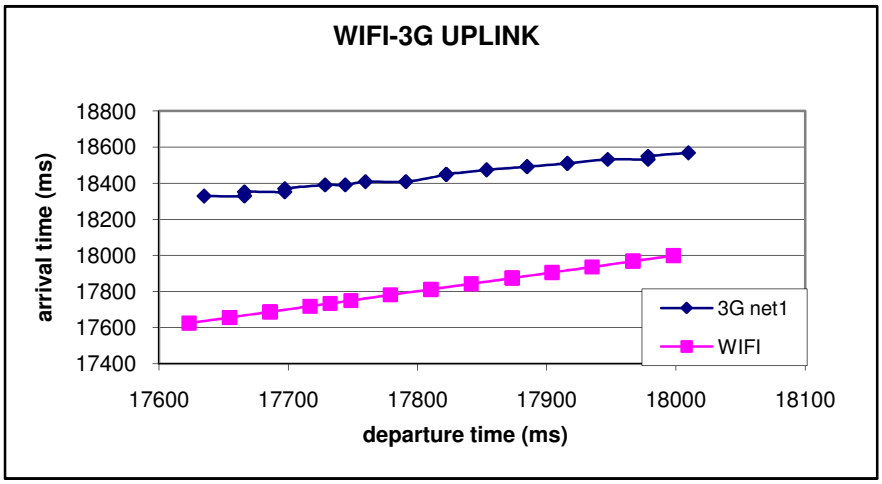
2.3.2 2^ measure: 2006-05-21, 3.25 pm

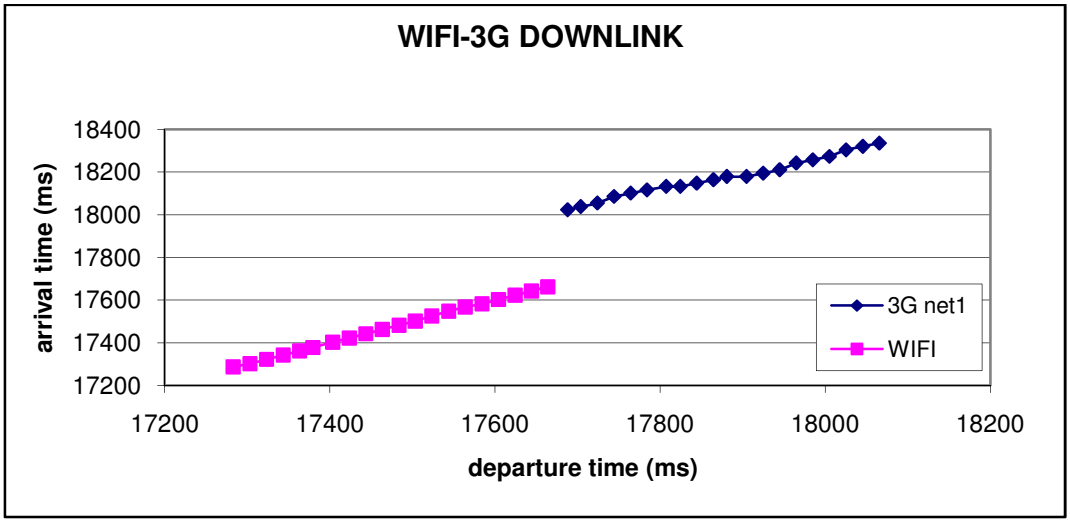




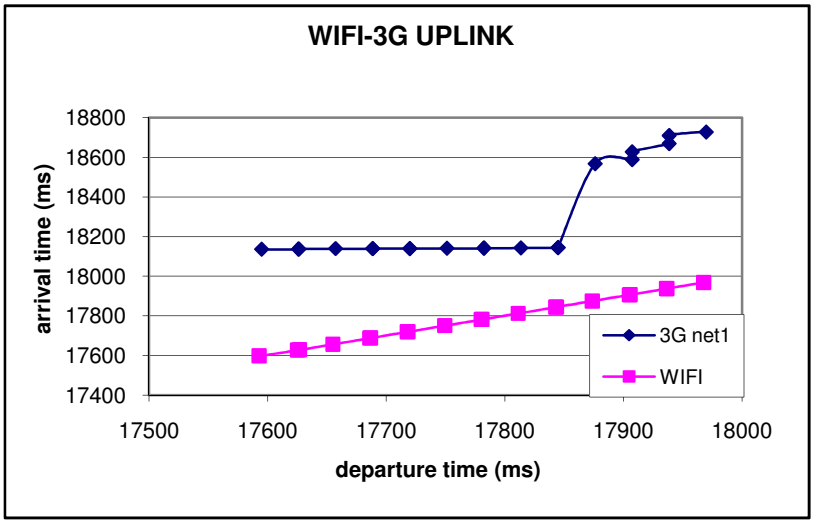
2.4 HANDOVER WIFI CAMPUS NETWORK – 3G (net1)

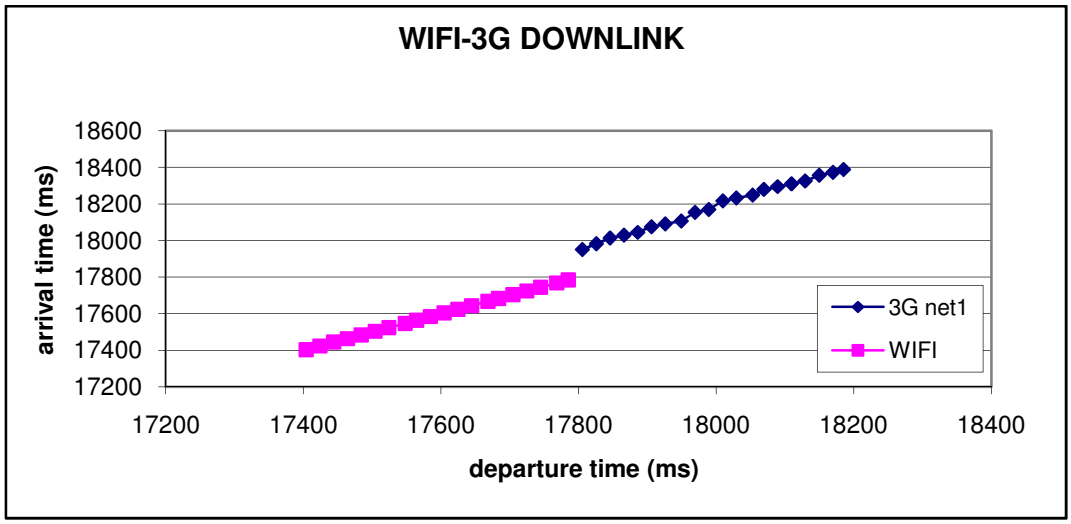
2.4.1 1^ measure: 2006-05-21, 2.30 pm





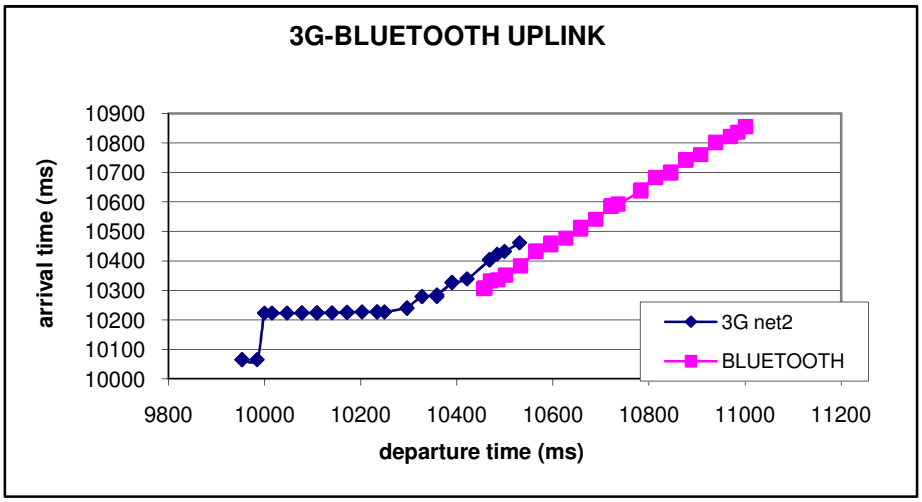
2.4.2 2^ measure: 2006-05-21, 4.00 pm

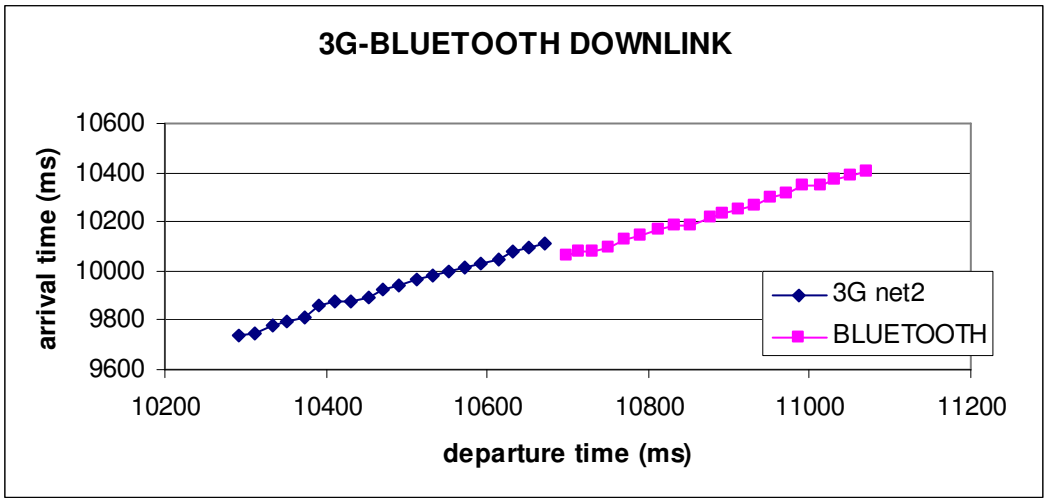




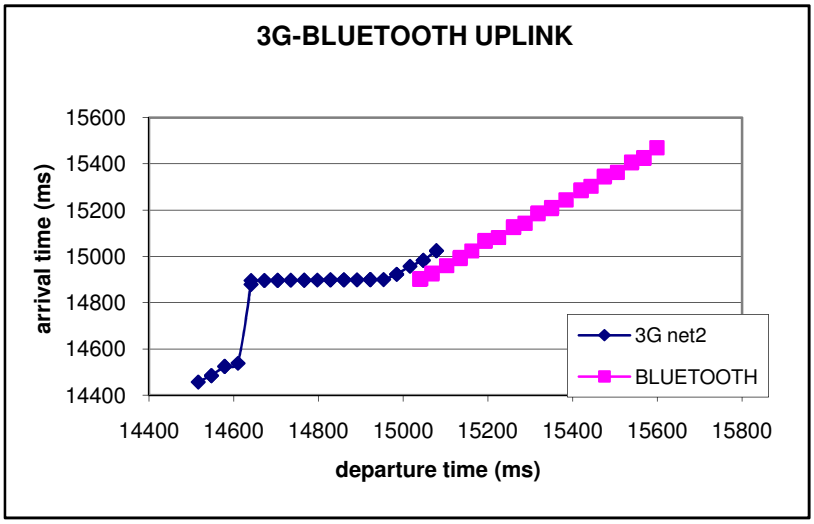
2.5 HANDOVER 3G (net2) - BLUETOOTH NETWORK

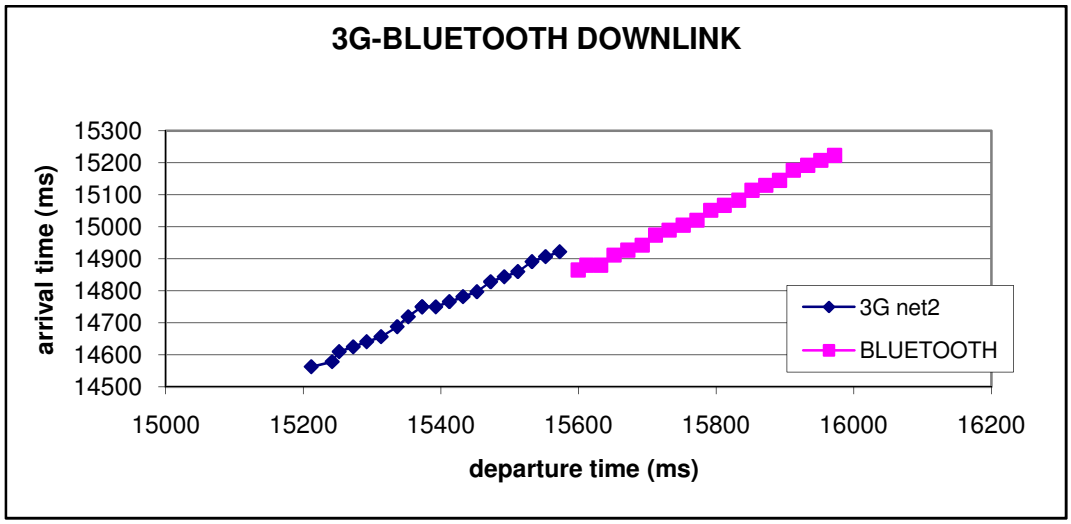
2.5.1 1^ measure: 2006-05-18, 4.00 pm





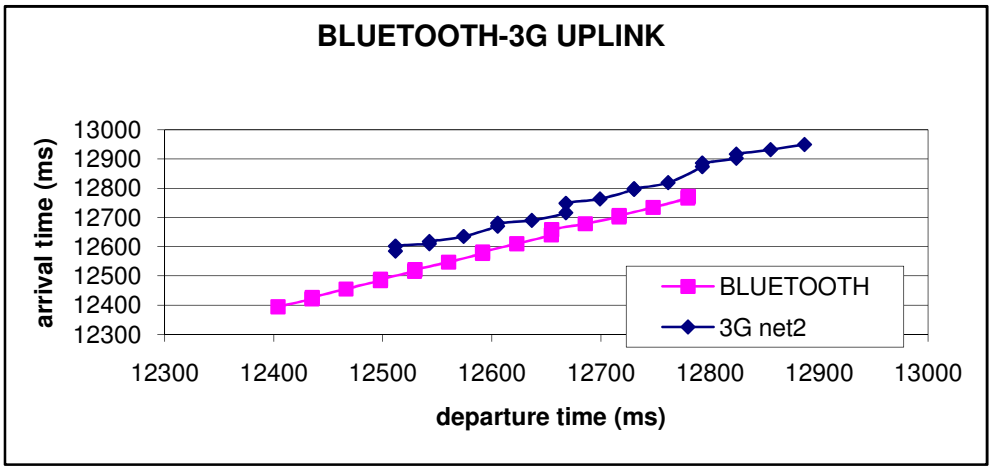
2.5.2 2^ measure: 2006-05-18, 5.00 pm

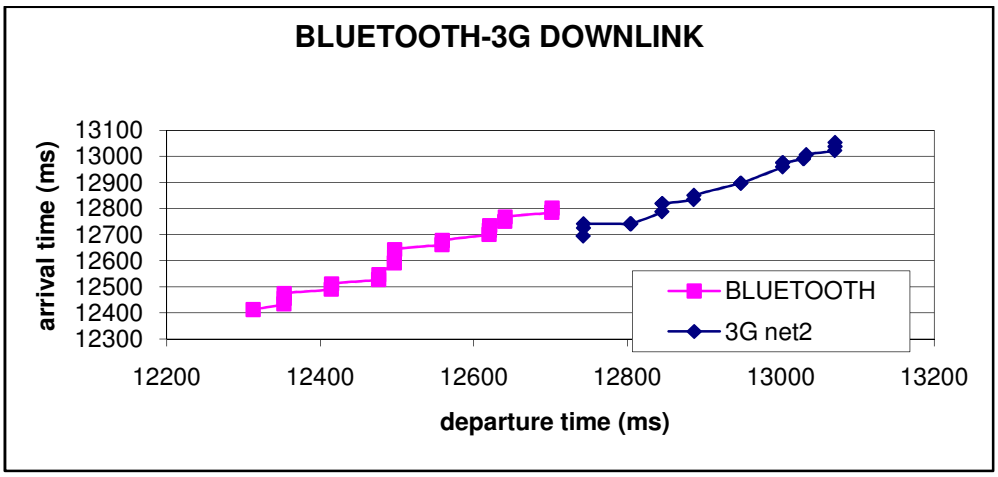




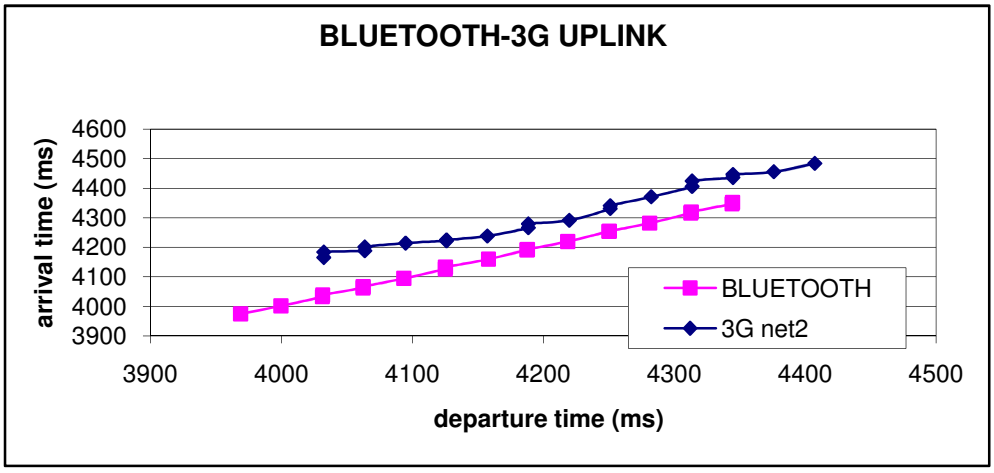
2.6 HANDOVER BLUETOOTH NETWORK – 3G (net2)

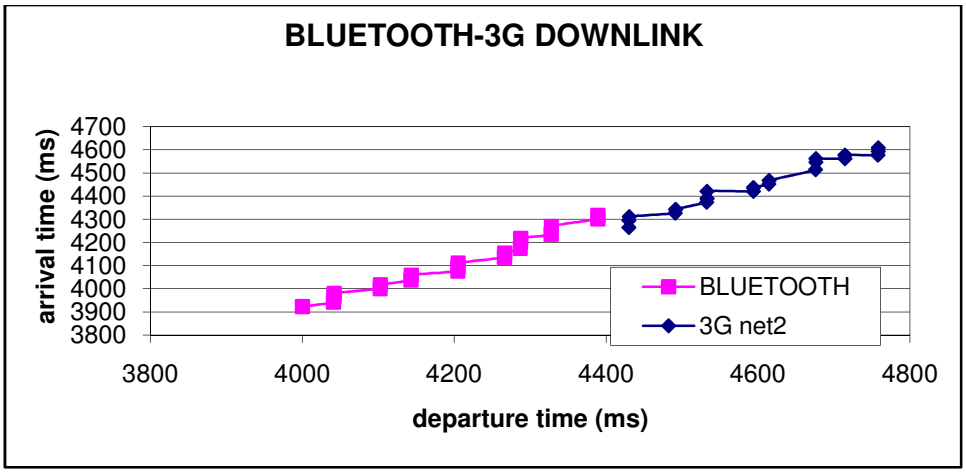
2.6.1 1^measure: 2006-10-20, 12.00 am





2.6.2 2^measure: 2006-10-20, 15.00 am

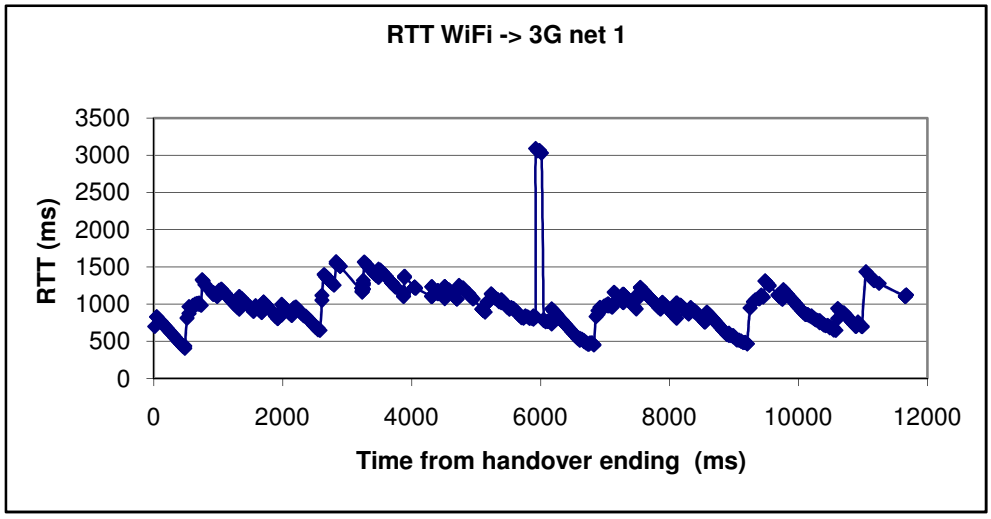




3 Evaluation of RTT in handover procedure

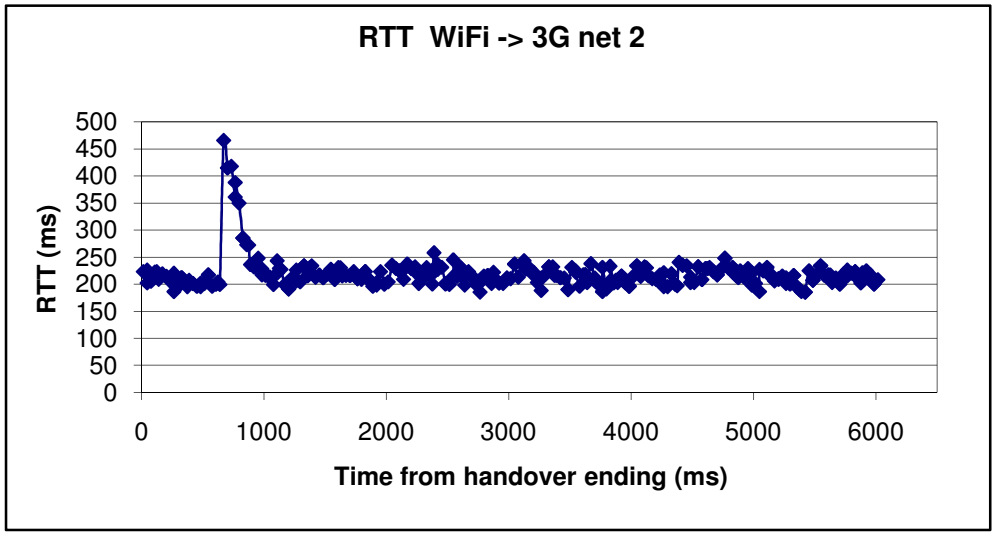
3.1 HANDOVER WIFI CAMPUS NETWORK – 3G (net1)

3.1.1 1^ measure: 2006-05-21, 2.30 pm



3.2 HANDOVER WIFI CAMPUS NETWORK – 3G (net2)

3.2.1 3^ measure: 2006-05-20, 12.50am



3.3 HANDOVER BLUETOOTH NETWORK – 3G (net2)

3.3.1 1^ measure: 2006-05-18, 10.00 am