

Architecture and testbed implementation of vertical handovers based on SIP session border controllers

Stefano Salsano · Luca Veltri ·
Andrea Polidoro · Alessandro Ordine

Received: 25 May 2006 / Accepted: 20 January 2007 / Published online: 31 March 2007
© Springer Science+Business Media B.V. 2007

Abstract This paper describes a Session Initiation Protocol (SIP) based solution for mobility management that provides seamless mobile multimedia services in a heterogeneous scenario where different radio access technologies are used (802.11/ WiFi, Bluetooth, 2.5G/3G networks). The solution relies on the so called “Session Border Controllers” which are now widely used in many commercial SIP telephony solutions, mainly to deal with NAT traversal. Session Border Controller functionality has been extended to support seamless mobility for multimedia applications. A prototype of the proposed solution focused on VoIP services has been implemented in a test bed which is able to perform seamless handovers (and NAT traversal) using the 802.11, Bluetooth and 3G (UMTS) access networks. Measurements results are reported which analyze the performance of the solution in a real world environment, using commercial WiFi and 3G services.

Keywords Seamless handover-SIP · Application layer mobility · Session Border Controller · WiFi · 3G networks

1 Introduction

Several different wireless access technologies are now available that can support real time services, in particular voice over IP (VoIP). Currently, these technologies span from cellular networks (UMTS), LAN technologies (802.11 a/b/g) and even PAN technology (Bluetooth), but other technologies will become mature in the very near future (e.g. 802.16). Multi standard terminals, laptops, tablet PCs, PDAs and even phones are now able to use more than one interface at a time. One of the goal of next generation mobile networks (often referred to as 4G network) is the integration and interoperability of different access technologies (including fixed access with “fixed mobile convergence”) into a single system or better into a single service presented to the user. The integration can happen at different levels of the protocol stack and various solutions are under study which operate at these different levels. In this paper we focus on an application level solution based on the SIP protocol [18,23]. The advantage of this type of solution is that it easily adapt to whatever underlying access technology is used.

The service scenario that we have considered is *wireless VoIP*. Nothing prevents to extend it to

S. Salsano (✉) · A. Polidoro · A. Ordine
Department of Electronic Engineering,
University of Rome “Tor Vergata” Via del Politecnico
1, 00133 Rome, Italy
e-mail: stefano.salsano@uniroma2.it

L. Veltri
Department of Information Engineering, University of
Parma, Parma, Italy

wireless multimedia (e.g. for video communication) provided that the access network(s) offer suitable capacity. Wireless VoIP service can be accessed over: (i) Local Area Networks (LAN) where voice communication are provided to a company over an enterprise wireless LAN (WLAN) or to people accessing public WLAN hot spots; (ii) Personal Area Networks (PAN) for example offering voice communication using Bluetooth technology in a domestic environment or in a small office; (iii) Wide Area Network (WAN) thanks to high data rate 2.5G/3G cellular technologies. The ability to use multiple networks in parallel gives the user a possibility to choose the most economical or the most performing access network at a given time, or conversely it gives the service provider the possibility to use the most suitable connection for each application. Therefore we face the requirement to support seamless mobility on multi-mode terminals, with the ability to place and receive calls over the most suitable wireless interface and to maintain VoIP sessions alive while handing off between the different access networks. As most of the wireless access networks are currently using private IP addresses and are connected through Network Address Translation (NAT) elements, the “NAT traversal” is one of the most critical issues to ubiquitously deploy VoIP services in the real world. The support of the “NAT traversal” combined with mobility and handover functionality is a very important requirement.

In this paper we describe a solution for this issue. The solution takes care of the “vertical mobility” of a user among different access networks/technologies, considering that for each different attached network the terminal will receive and use a different IP address. On the other hand, the movement of the user among base stations of the same technology/network (e.g. among different access points in the same WiFi campus, or among different 3G cells in the cellular network of an operator) is handled by specific mechanisms of the given access network and no IP re-configuration is required.

In our solution, we consider a Mobile Terminal (MT) with multiple network interfaces that can be active at the same time making it possible to realize a “soft handover”. As an example Fig. 1 illustrates the scenario of a handover between a WiFi and a 3G network (this scenario is a subset of what has

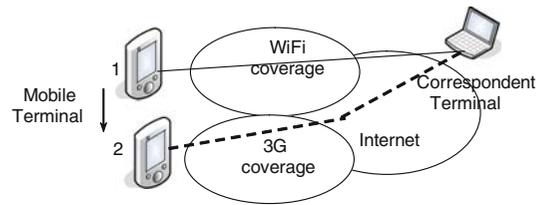


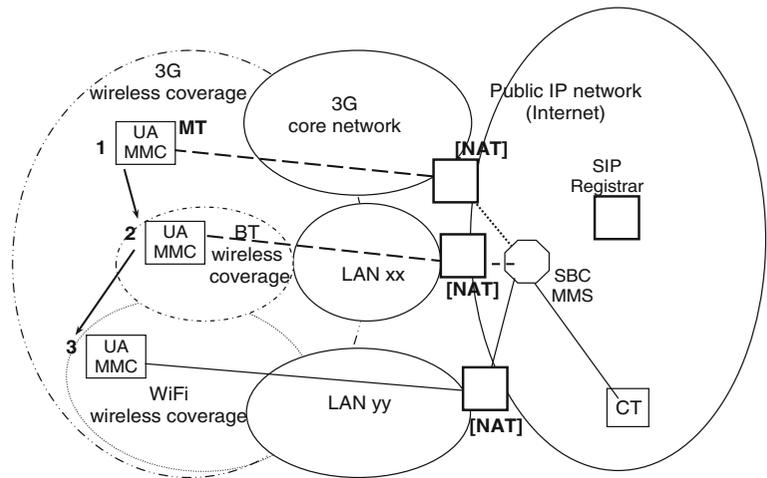
Fig. 1 VoIP handoff scenario in a WiFi/3G interworking scenario

been also implemented in our testbed that will be described later on). The Session Initiation Protocol (SIP) will be used for both “traditional” VoIP signaling and for supporting terminal mobility. The solution integrates mechanisms to enable MTs to make and receive VoIP calls regardless if they are located inside a public or private IP network.

The paper is organized as follows. We introduce the main concepts of the proposed solution (Sect. 2) and then discuss the details of the signaling procedures (Sect. 3). Section IV describes our testbed and the achieved measurement results, while Sect. 5 provides a report of existing work linked to our solution.

2 Proposed solution

The fundamental concepts of the proposed solution can be illustrated with the help of Fig. 2. MTs have access to different networks (in the figure, WiFi, Bluetooth and cellular 3G 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. 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) [8] 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 MT 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

Fig. 2 Architecture

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, as shown in the figure. Obviously the MMS and SIP Registrar can be implemented in a single element if required.

2.1 From SIP SBCs to MMS

While a large part of the target terminals are using private IP addresses and are “hidden” behind a NAT, signaling protocols for VoIP, like SIP, do not “natively” support full NAT traversal. NAT traversal mechanisms are needed to allow terminals in “private” networks to be reached (i.e. “invited” to a phone call), and to allow the media streams to be setup between the caller and the called UAs, notwithstanding NAT boxes that could be placed at the borders between the private and the public network. IPv6 promises to overcome the NAT issues but wide spread diffusion of IPv6 is not foreseen in the short/medium term. Moreover, in future IPv6 networks, the problem of address/port mapping may still be present due to some restrictions introduced for security reasons. A SIP SBC 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 more 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 [17,20], 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.

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.

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 CT (and other SIP intermediate nodes) completely unaware of what is occurring. Instead, in the traditional SIP approach the CT is directly involved by the MT with a new “Re-INVITE” transaction. In such case, the CT is in charge of performing the handoff by establishing a new media flow using the IP address of the terminal in the “target” network. A first advantage of the SBC based handover with respect to CT-based handover is that the solution does not rely on the CT capability to perform the handover. Though the SIP standard requires that terminals are able to support Re-INVITE, compatibility issues may arise and it will be difficult to verify that the handover works with all possible SIP terminals. Moreover executing the handover with the CT could lead to high delays, for this reason various solutions have been proposed trying to overcome this problem by introducing intermediate entities as temporary anchor points (see [1, 10]). In our solution the anchor point is not temporary, but permanent. This permanent media relay in the path is not a very efficient solution, but in practice we witness several running services that are implemented in this way. It is also worth noting that the only other solution which allows to overcome a “symmetric NAT” is using TURN [17], but this exactly requires a media proxy box that acts as permanent media relay (and it introduces greater setup and handover latency respect to a SBC-based solution). If we can assume

that a SBC is already in the path the introduction of a media relay is definitely not a shortcoming introduced by our solution. We are simply extending the SIP and media processing capability of the SBC in order to support the handover.

Note that the proposed solution can be applied exactly as described both in networks with private IP addresses and in networks with public IP addresses. In a scenario with private IP networks and in a mixed scenario (where the MT can roam among both networks with private IP addresses and networks with public IP addresses) the proposed solution is able to solve the NAT traversal issues together with the mobility management. If we consider a scenario where the terminal is only roaming among networks with public IP addresses, we lose one of the advantages of our solution (as no NAT traversal is needed). This scenario does not seem realistic in current networks. Anyway, our solution could also prove useful in this context, due to the other advantages listed above: the handover procedure does not rely on the remote terminal, the handover delay could be better controlled. Moreover for VoIP services offered by a public network operator, the need to provide “lawful interception” could imply that media flows are conveyed in any case through a media relay.

For the sake of simplicity, we only describe the solution considering a single centralized SBC. In real life, redundancies must be considered to achieve reliability (e.g. the SBC needs to be replicated). Likewise, the SBC functionality could need to be distributed for scalability reasons. A possible interesting approach is to split the signaling and media relay capability of the SBC (see for example [14]), which allows to deploy a set of media relay boxes that can be placed “close” to the MT. Replication mechanisms for reliability and distribution mechanisms for scalability are out of the scope of this paper.

2.2 The MMC in the MT

The MMC can be implemented as shown in Fig. 3 as a separate entity running on the MT that masquerades all mobility and NAT traversal functionality by relaying both signaling and media flows. In this case the SIP User Agent sees the MMC as

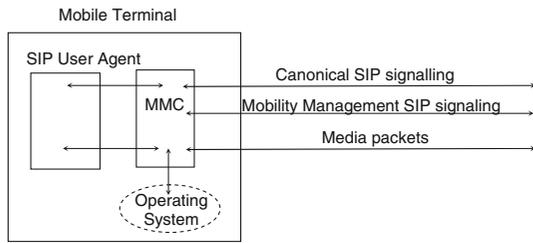


Fig. 3 The MMC in the MT

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. A different solution would be to integrate the MMC functionality within the UA, which will likely imply a greater efficiency in the use of processing resource of the MT. The two solutions only differ in the internal implementation, while there is no difference in the external behavior of the procedures. In our test-bed we used the first solution (the one depicted in Fig. 3) so that we can use any existing SIP User Agent.

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). The discussion of these criteria are out of the scope of this paper.

3 Specification of the 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.

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) [12] bodies in order to force itself as relay for the media packets.

In the SIP architecture, the SIP Registrar records the current location(s) of the user. An SBC coupled with an external backend SIP Registrar introduces a two levels user location mechanism, where the SIP user address (called “Address of Record” or AoR) is mapped by the Registrar server to a new user Uniform Resource Locator (URL) referring to the SBC and then the SBC maps this new URL to the actual address of the user’s UA. The presence of a backend Registrar server can let the overall architecture be more flexible and compatible with other server-based services, leaving the complete control of such services (e.g. AoR resolution or Call Processing Language (CPL)-based services) to the backend Registrar/Proxy. Following this approach, in our proposal the mobility of the terminal amongst different access networks is controlled by the MMS/SBC, while the external SIP Registrar simply points to the MMS/SBC of each registered user (this means that the user’s AoR is simply mapped to a user-specific URL pointing to the MMS/SBC). The SBC/MMS will take care of NAT traversal, so that the MT can be reached by SIP signaling and can send/receive media flows even beyond a NAT. As for SIP signaling, the MMC in the MT and the MMS in the SBC implement the SIP extension described in [19] which allows the MMC to receive SIP responses on the same port where it sent corresponding SIP requests. A “keep-in-touch” mechanism is needed to keep the pinhole in the NAT open. Various techniques can be used [18] such as dummy UDP packets (from the MMC to the MMS or vice-versa), malformed SIP messages, well-formed SIP messages. We use periodic SIP register messages from MMC to MMS. The “keep-in-touch” packets are sent every 30 s, so they use a very limited amount of resources.

3.1 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 Fig. 4.A. The MMC in the MT sends a Registration Request to the MMS over the “selected” interface. 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).

As result of the Location Update Registration procedure, 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. Several solutions are possible and a detailed analysis of this issue cannot be included in this paper for space constraints. We used an identifier that the MMC inserts in the Contact and in the Via header fields, and it is denoted as Terminal-ID in the SIP messages shown in Fig. 5. The details of the solution have used in the test-

bed can be found in [11], which reports the complete SIP messages related to the various procedures.

3.2 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 Fig. 4B. 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. 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. In message M5 of Fig. 5 there is an example of the rewritten contact (further details can be found in [11]). 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.

3.3 Session establishment

The session establishment procedure consists in a standard SIP session setup procedure. All session establishment messages for MT are handled by the 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

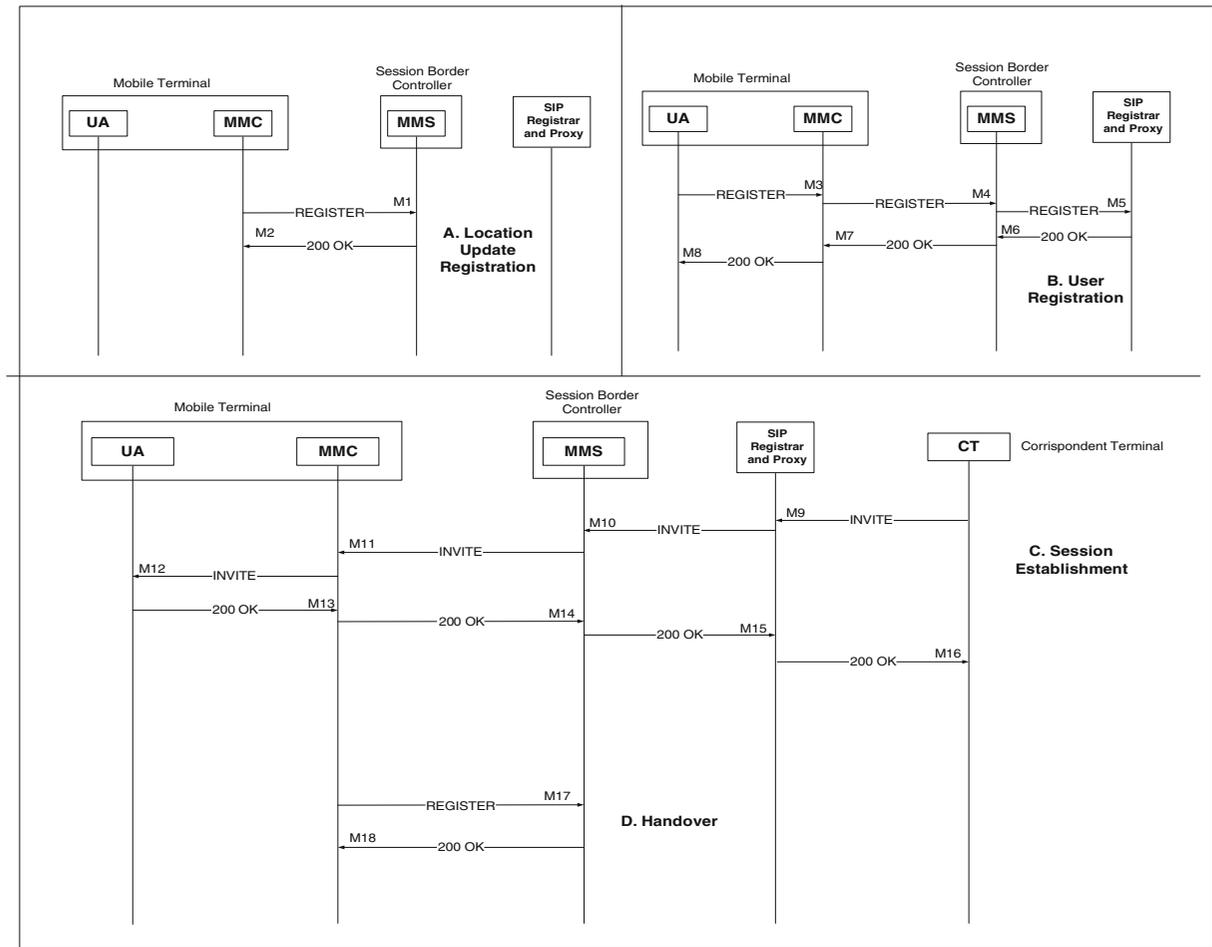


Fig. 4 Signaling procedures

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. 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 the received keep-alive packets so that they are not forwarded to the CT.

3.4 On-call mobility: the handover procedure

The on-call mobility management procedure takes place when the UA identifies the need for hand-off during an ongoing VoIP 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 is MT initiated. The MMC in the terminal sends an

<p>Message 1: MMC to MMS</p> <p>REGISTER sip:MMS_IP SIP/2.0 Via: SIP/2.0/UDP Terminal_ID; branch=z9h To: <User_ID> From: < User_ID >;tag=dba Call-ID: 7bb@002 CSeq: 1 REGISTER Contact: <sip: Terminal_ID></p>	<p>Message 2: MMS to MMC</p> <p>SIP/2.0 200 OK Via: SIP/2.0/UDP Terminal_ID;branch=z9h; received=IP_NAT To: <User_ID> From: < User_ID >;tag=dba Call-ID: 7bb@002 CSeq: 1 REGISTER</p>	<p>Message 3: UA to MMC</p> <p>REGISTER sip:domain.net SIP/2.0 Via: SIP/2.0/UDP MMC_IP;rport;branch=z9h From: <sip:user@domain.net>;tag=25 To: <sip:user@domain.net> Contact: <sip:User_ID@MMC_IP> Call-ID: FDA@domain CSeq:1 REGISTER</p>
<p>Message 4: MMC to MMS</p> <p>REGISTER sip:domain.net SIP/2.0 Via: SIP/2.0/UDP Terminal_ID;branch=z9h; Via: SIP/2.0/UDP MMC_IP;rport;branch=z9h From: <sip:user@domain.net>;tag=25 To: <sip:user@domain.net> Contact: <sip: Terminal_ID> Call-ID: FDA@domain CSeq: 1 REGISTER</p>	<p>Message 5: MMS to Registrar</p> <p>REGISTER sip:Reg_IP SIP/2.0 Via: SIP/2.0/UDP MMS_IP;branch=z9h Via: SIP/2.0/UDP Terminal_ID;branch=z9h; Via: SIP/2.0/UDP MMC_IP;rport;branch=z9h Route: <sip: Reg_IP;r> From: <sip:User_ID@REG_IP>;tag=157 To: <sip:User_ID@REG_IP> Call-ID: FDA@domain CSeq: 1 REGISTER Contact: <sip/MMS_ID-Terminal_ID@MMS_IP></p>	<p>Message 6: Registrar to MMS</p> <p>SIP/2.0 200 OK Via: SIP/2.0/UDP MMS_IP;branch=z9h Via: SIP/2.0/UDP Terminal_ID;branch=z9h; Via: SIP/2.0/UDP MMC_IP;rport;branch=z9h From: <sip:User_ID@REG_IP>;tag=157 To: <sip:User_ID@REG_IP> Call-ID: FDA@domain CSeq: 1 REGISTER</p>
<p>Message 7: MMS to MMC</p> <p>SIP/2.0 200 OK Via: SIP/2.0/UDP Terminal_ID;branch=z9h; Via: SIP/2.0/UDP MMC_IP;rport;branch=z9h From: <sip:user@domain.net>;tag=25 To: <sip:user@domain.net> Call-ID: FDA@domain CSeq: 1 REGISTER</p>	<p>Message 8: MMC to UA</p> <p>SIP/2.0 200 OK Via: SIP/2.0/UDP MMC_IP;rport;branch=z9h From: <sip:user@domain.net>;tag=25 To: <sip:user@domain.net> FDA@domain CSeq:1 REGISTER</p>	<p>Message 9: CT to Proxy</p> <p>INVITE sip:/MT_Terminal_ID@MMS_IP SIP/2.0 Via: SIP/2.0/UDP CT_IP; branch=z9h From: <sip:CTuser@domain.net>;tag=871 To: <sip:MTuser@domain.net> Call-ID: F16@192 CSeq: 1 INVITE Contact: <sip:CT_contact> Content-Length: 214</p>
<p>Message 10: Proxy to MMS</p> <p>INVITE sip:/ MMS_ID-MT_Terminal_ID@MMS_IP SIP/2.0 Via: SIP/2.0/UDP Proxy_IP;branch=z9h Via: SIP/2.0/UDP CT_IP From: <sip:CTuser@domain.net>;tag=871 To: <sip:MTuser@domain.net> Call-ID: F16@192 CSeq: 1 INVITE Contact: <sip:CT_contact> Content-Length: 214</p>	<p>Message 11: MMS to MMC</p> <p>INVITE sip:/ MT_Terminal_ID@ SIP/2.0 Via: SIP/2.0/UDP MMS_IP;branch=z9h Via: SIP/2.0/UDP Proxy_IP;branch=z9h Via: SIP/2.0/UDP CT_IP From: <sip:CTuser@domain.net>;tag=871 To: <sip:MTuser@domain.net> Record-Route: <sip:MMS_IP;r> Call-ID: F16@192 CSeq: 1 INVITE Contact: <sip/MMS_ID/-CT_ID/AT- MMS_IP/PORT-5070@MMS_IP> Content-Length: 214</p>	<p>Message 12: MMC to UA</p> <p>INVITE sip:/ MT_Terminal_ID SIP/2.0 Via: SIP/2.0/UDP MT_Terminal_ID;branch=z9h Via: SIP/2.0/UDP MMS_IP;branch=z9h Via: SIP/2.0/UDP Proxy_IP;branch=z9h Via: SIP/2.0/UDP CT_IP From: <sip:CTuser@domain.net>;tag=871 To: <sip:MTuser@domain.net> Call-ID: F16@192 CSeq: 1 INVITE Contact: <sip/MMS_ID/-CT_ID/AT- MMS_IP/PORT-5070@MMS_IP> Content-Length: 214</p>
<p>Message 13: UA to MMC</p> <p>SIP/2.0 200 OK Via: SIP/2.0/UDP MT_Terminal_ID;branch=z9h Via: SIP/2.0/UDP MMS_IP;branch=z9h Via: SIP/2.0/UDP Proxy_IP;branch=z9h Via: SIP/2.0/UDP CT_IP From: <sip:CTuser@domain.net>;tag=871 To: <sip:MTuser@domain.net> Call-ID: F16@192 CSeq: 1 INVITE Contact: < sip:MT_User_ID@MMC_IP ></p>	<p>Message 14: MMC to MMS</p> <p>SIP/2.0 200 OK Via: SIP/2.0/UDP MMS_IP;branch=z9h Via: SIP/2.0/UDP Proxy_IP;branch=z9h Via: SIP/2.0/UDP CT_IP From: <sip:CTuser@domain.net>;tag=871 To: <sip:MTuser@domain.net> Call-ID: F16@192 CSeq: 1 INVITE Contact: < sip:MT_User_ID@MMC_IP ></p>	<p>Message 15: MMS to Proxy</p> <p>SIP/2.0 200 OK Via: SIP/2.0/UDP Proxy_IP;branch=z9h Via: SIP/2.0/UDP CT_IP From: <sip:CTuser@domain.net>;tag=871 To: <sip:MTuser@domain.net> Call-ID: F16@192 CSeq: 1 INVITE Contact: < sip:MT_User_ID@MMC_IP ></p>
<p>Message 16: Proxy to CT</p> <p>SIP/2.0 200 OK Via: SIP/2.0/UDP CT_IP; branch=z9h From: <sip:CTuser@domain.net>;tag=871 To: <sip:MTuser@domain.net> Call-ID: F16@192 CSeq: 1 INVITE Contact: < sip:MT_User_ID@MMC_IP ></p>	<p>Message 17: MMC to MMS</p> <p>REGISTER sip:MMS_IP SIP/2.0 Via: SIP/2.0/UDP Terminal_ID; branch=z9h To: <User_ID> From: < User_ID >;tag=dba Call-ID: 7bb@002 CSeq: 1 REGISTER Contact: <sip: Terminal_ID> HO_Call_Id:F16@192</p>	<p>Message 18: MMS to MMC</p> <p>SIP/2.0 200 OK Via: SIP/2.0/UDP Terminal_ID;branch=z9h; received=IP_NAT To: <User_ID> From: < User_ID >;tag=dba Call-ID: 7bb@002 CSeq: 1 REGISTER</p>

Fig. 5 SIP messages

“handover” Register message over the target network interface addressed to the MMS in the SBC. Differently from a Location Update Register request, the handover Register request contains in the message body the reference to the active session 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 in the SBC receives the Register message, it will start accepting packets coming from the new interface and discarding the ones coming from the old interface for the media flows corresponding to the call ID contained in a specific header in the handover Register message. Then it will send back the SIP 200 OK message to the MMC and start 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” Register message 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 Register if the 200 OK is not received back. The SIP standard suggests a default value of the retransmission timeout equal to 500ms, that is doubled on each retransmission. However this is not compatible with a reasonable performance of the handover in case of the loss of the Register message. Therefore we mandate that for the “handover” Register message a different duration of the retransmission timer is used. Register messages are sent with a fixed interval of 200ms until the 200 OK is received or a transaction timeout occurs. The transaction timeout is set to 3s corresponding to a maximum of 15 retransmissions. On the terminal 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.

The criteria for taking the handover decision can be based on:

- the quality of the received signal (power, S/N ratio)
- on IP level measurements of the QoS in the path between the MT and the SBC over the different wireless networks (packet loss, jitter)
- on the cost of the connections.

The analysis of these criteria is out of the scope of this paper.

3.5 Comparison with canonical SIP based mobility

In classical SIP based mobility, when the MT moves to a new access network changing its IP address during a call it re-invites the remote CT in order to re-establish a new media streams (the handover is handled by the remote CT). Then the MT has to register the new address to the SIP registrar. Instead, in the proposed solution the two functions are tied in just one registration procedure between the MT and the SBC, while the corresponding terminal is let completely unaware of the MT movement. This increases the handover performances, increases the compatibility with legacy remote terminals that might not handle correctly the re-invitation procedure, guarantees a better privacy (since the position and movement of the MT are hidden by the SBC).

4 Implementation and measurements

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 Fig. 6. The MTs 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 [15]. As SIP User Agent we used

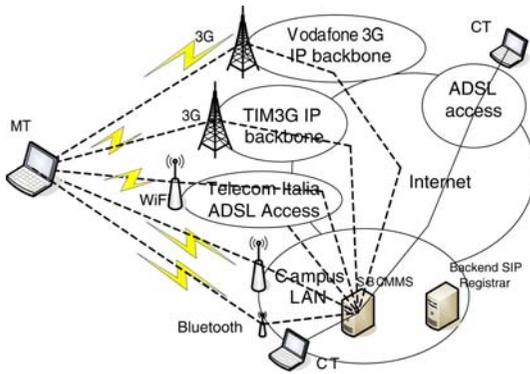


Fig. 6 Testbed layout

the Xlite software client [24]. 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 [3]. 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.

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 appli-

cation (Sect. 4.1) and then the performance of the proposed handover procedure (Sect. 4.2). 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 Fig. 8), 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 [6] based upon a reduction of the ITU-T’s E-Model [13] to transport level measurable quantities

4.1 Evaluation of access network performance

In order to evaluate the performance of different access networks, we have developed a tool named “Throcal” 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.

A more detailed report of the measurement campaign can be found in [11], we only present here the main results. We measured a very good (i.e. low) loss rate using all the different access network. Table 1 reports a sample of our loss measurements over the different access network. On the

Table 1 Packet loss ratio of the different network access

	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%

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). Fig. 7 shows the cumulative distribution of RTT for the different access networks. Each distribution is evaluated with five different tests of duration 60s repeated with 15 min interval during a working hour (e.g. 11 am). The average RTT is in the order of 400 and 800 ms for the two networks and even worse is the 95% percentile which is in the order of 600 and 1,000 ms, resulting in a degradation of voice experience.

4.2 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 [7]. 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 Fig. 8, 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 us define as U_{up} and U_{dn} 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 Fig. 6). Similarly we define W_{up} and W_{dn} for the WiFi network. In the performed experiments, the measured RTT between the MT and the SBC for the 3G access (i.e. $U_{up}+U_{dn}$) was in the range of 200 ms (as shown in Fig. 9), while for the WiFi access (i.e. $W_{up}+W_{dn}$) was in the range of 20–25 ms.

Figure. 8 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. $U_{up} - W_{up}$). 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 80–110 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 Fig. 8a, 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 ($U_{up} - W_{up}$). As for the handover completion time, it roughly corresponds to the RTT on the target interface (3G in this case: $U_{up} + U_{dn}$).

We measured 270 ms for the interval between the REGISTER and the 200 OK (i.e. the handover

Fig. 7 Cumulative distribution of RTT for different access network

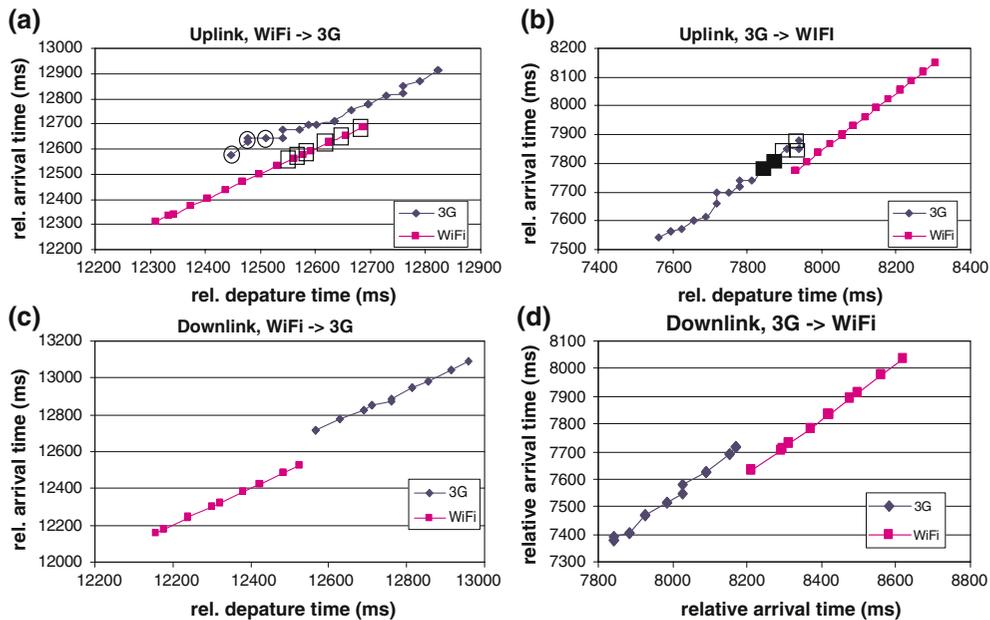
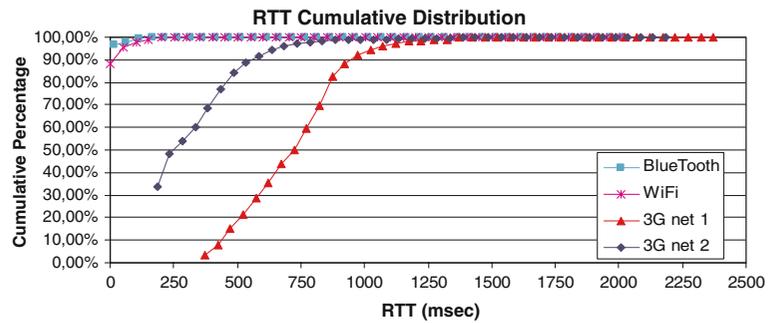


Fig. 8 RTP arrival patterns during handovers in 4 scenarios

duration as perceived by the MT) in the test shown in Fig. 8a. As a confirmation, we can see that in Fig. 8a the packets are duplicated from $t = 12,430$ ms to $t = 12,700$ ms. This duration is almost entirely caused by the RTT between MT and SBC/MMS. This is confirmed in Fig. 9 which shows the RTT measured analyzing the traces of RTP packets captured on the MT and on the SBC/MMS for 6 s following the handover.

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 Fig. 8b). 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 Fig. 8b. 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 Fig. 8b. In fact, the packets are duplicated from $t = 7,906$ ms to $t = 7,938$ ms. Coming to the downlink flows, let us consider the handover from WiFi to 3G (Fig. 8c). The SBC will

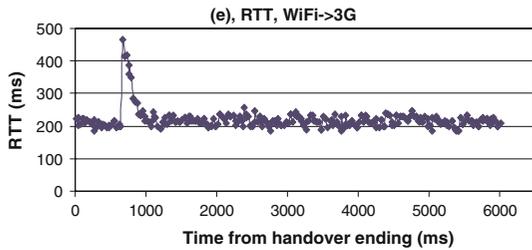


Fig. 9 RTT estimation after handover procedure ending (3G network)

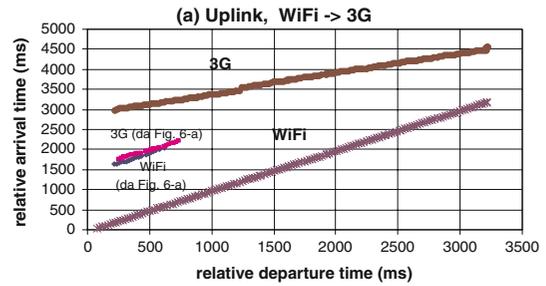


Fig. 11 RTP arrival pattern without keep-alive on the unselected 3G interface

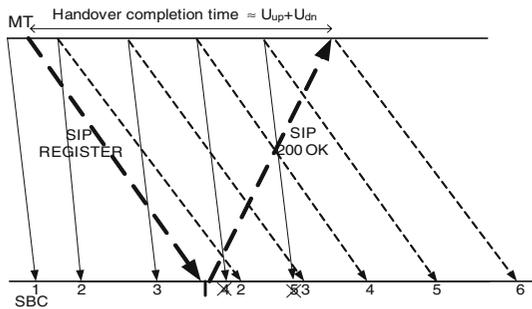


Fig. 10 Temporal diagram for WiFi → 3G handover

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 ($U_{dn}-W_{dn}$) which is in the order of 200 ms in Fig. 8c. 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 (Fig. 10).

Finally, let us consider the handover from 3G to WiFi for the downlink flow (Fig. 8d). 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 200 ms in Fig. 8d. Note that no packets are lost in this handover, the gap in Fig. 8d 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 Fig. 8c and d we can evaluate the difference in one-way delay for the downlink $U_{dn}-W_{dn}$. In our test we measured that 3G one-way downlink delay was from 80 ms 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 $U_{up}-W_{up}$ and $U_{dn}-W_{dn}$ are in the order of 110 ms, the handovers are not perceived.

It is interesting to compare the results shown in Fig. 8 with the corresponding measurements without using the keep-alive mechanism introduced in Sect. 3.3. As we can see in Fig. 11, 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 Fig. 8a in an arbitrary position in the left part of Fig. 11.

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 Fig. 8 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 Fig. 10 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 $W_{\text{up}} + W_{\text{dn}}$ for the handover $3\text{G} \rightarrow \text{WiFi}$ and in the order of $W_{\text{up}} + U_{\text{up}}$ for the handover $\text{WiFi} \rightarrow 3\text{G}$.

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. 1 and in Fig. 7. To support this statement, in [11] 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.

5 Related work

A survey on the different mobility management approaches to support mobility in VoIP services can be found in [2]. Mobility mechanisms for IP networks can be classified in IP network layer, transport layer or on application layer mechanisms. Mobility is provided at IP layer by the Mobile IP (MIP) [16]. Mobile IP is not directly related to VoIP applications, since it is completely transparent to the upper layers performing the same behavior for all incoming/outgoing IP datagrams. It provides the MT with a single IP address that “follows” the terminal in its wandering. The main disadvantage of MIP approach is that it needs a

support from the routers in each access network. A MIP terminal will be able to roam only over MIP enabled access network.

SIP-based solutions belong to application layer mechanisms and they are usually based on SIP signaling for re-negotiating the media sessions when a handover occurs, as described also in [23]. In the classical SIP based approach the handover is performed end-to-end, only involving the MT and the CT.

In both the approaches (MIP and SIP) there is the need to minimize the service disruption during a handoff procedure and this issue has been addressed in several works (see for example [5, 10]). An interesting work on SIP-based mobility can be found in [1]. The approach proposed in [1] is similar to the one exploited by our proposal, with some fundamental differences. In [1] the authors propose to re-negotiate the session with the remote UA (assisted by an intermediate SIP node) while we propose to use only the intermediate node (the SBC/MMS), with the main advantages of: (i) reducing the latency of the overall handover procedure (ii) overcoming eventual compatibility problems introduced by legacy remote UAs—in fact, in our proposal the remote UA is completely unaware of the mobility procedure. Note that the latter aspect has also some useful security implication (i.e. the remote UA can not trace the complete movement and the current position of the MT). Another important difference is related to the mode of operation of the intermediate node that in our solution is proxy-like instead of the B2BUA model proposed in [1]. We think that a proxy-style behavior is more flexible, less processor consuming, introduces less latency, and reduces the possibility of signaling incompatibility.

Other interesting works address the issue of 3G/WLAN interworking/integration, both in the research community (see for example [4, 21]) and in standardization for a like 3GPP or 3GPP2. The underlying idea is to include WLAN access in the set of services provided by 3G operators to their subscribers. Most of the work done focused on the authentication and security aspects, while the issues related to handoff are still to be investigated.

In our work we assume that the terminal is able to authenticate (separately or in an integrated manner) to the different access networks and we

focus on the problem of seamless mobility amongst heterogeneous networks and vertical handoff management. Similar focus can be found in [9], which considers the classical SIP based mobility (the handoff is handled by the correspondent terminal) rather than our solution to enhance the SBC.

A first version of this work has appeared in [22], which has been enhanced as follows: the role of SBC has been discussed, the details of SIP signalling have been provided, the Bluetooth PAN access network and the second 3G operator has been added to the testbed, additional performance results are included.

6 Conclusions and future work

In this paper we have presented a solution for seamless vertical handover between heterogeneous networks like WiFi and 3G, based on SIP. The novelty of the solution is that it is strongly coupled with the NAT traversal features provided by the so called SBCs. Assuming that the MTs will be mainly roaming on networks with private IP numbering, the mediation of an SBC is already proposed to solve the NAT traversal issue, therefore we straightforwardly propose to enhance SBC functionality to support the 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 4G 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 been run in order to provide quantitative evaluation of the roaming solution.

Ongoing work (which was not presented here for space constraints) concerns the realization of the mechanisms to drive the handover decision (both collecting the signal quality information from the network interface cards and making IP level measurements during the call active phase).

References

1. Banerjee, N., Acharya, A., & Das, S. K. (2006). Seamless SIP-based mobility for multimedia applications. *IEEE Network*, 20(2), 6–13.
2. Banerjee, N., Wu, W., & Das, S. K. (2003). Mobility support in Wireless Internet. *IEEE Wireless Communications*, 10(5), 54–61.
3. BlueZ, Official Linux Bluetooth protocol stack, <http://www.bluez.org/>
4. Buddhikot, M. M., Chandranmenon, G., Han, S., Lee, Y.-W., Miller, S., & Salgarelli, L. (2003). Design and implementation of a WLAN/CDMA 2000 interworking architecture. *IEEE Communication Magazine*, 41(11), 90–100.
5. Campbell, A. et al. (2000). Design, implementation and evaluation of cellular IP. *IEEE Personal Communications Magazine*, 7, 42–49
6. Cole, R. G., & Rosenbluth, J. H. (2001). Voice over IP performance monitoring. *ACM SIGCOMM Computer Communication Review*, 31(2), 9–24
7. Combs, G. et al. Ethereal: A network protocol analyzer. <http://www.ethereal.com/>
8. Cumming, J. Sip Market Overview, <http://www.dataconnection.com/network/download/whitepapers/sipoverview.pdf>
9. Dutta, A., Kim, B., Zhang, T., Baba, S., Taniuchi, K., Ohba, Y., & Schulzrinne, H. (2005). Experimental analysis of multi interface mobility management with SIP and MIP. *International conference on wireless networks, communications and mobile computing*, 13–16 June 2005, Vol. 2, pp. 1301–1306.
10. Dutta, A., Madhani, S., Chen, W., Altintas, O., & Schulzrinne, H. (2004). Fast-handoff schemes for application layer mobility management. PIMRC, Spain.
11. Fiorani, M., Labella, A., Ordine, A., Polidoro, A., Salsano, S., & Veltri, L. (2006). 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
12. Handley, M., & Jacobson, V. (1998). SDP: Session Description Protocol. IETF RFC 2327, April 1998.
13. ITU-T Recommendation G.107 (1998). The E-Model, a computational model for use in transmission planning.
14. MediaProxy, <http://mediaproxy.ag-projects.com/REAMDME>
15. Mjisp open source JAVA SIP stack, <http://www.mjsip.org>
16. Perkins, C. et al. Mobility support of IPv4. RFC 3344.
17. Rosenberg, J. (2006). Obtaining relay addresses from simple traversal underneath NAT (STUN). draft-ietf-behave-turn-02 (work in progress), October 2006.
18. Rosenberg, J. et al. (2002). SIP: Session Initiation Protocol. IETF RFC 3261, June 2002.
19. Rosenberg, J., & Schulzrinne, H. (2003). An Extension to the Session Initiation Protocol (SIP) for symmetric response routing. IETF RFC 3581, August 2003.
20. Rosenberg, J., Weinberger, J., Huitema, C., & Mahy, R. (2003). STUN-simple traversal of User Datagram Protocol (UDP) Through Network Address Translators (NATs). IETF RFC 3489, March 2003

21. Ruggeri, G., Iera, A., & Polito, S. (2005). 802.11-based wireless-LAN and UMTS interworking: Requirements, proposed solutions and open issue. *Computer Networks*, 47(2), 151–166.
22. Salsano, S., Veltri, L., Martiniello, G., & Polidoro, A. (2006). Seamless vertical handover of VoIP calls based on SIP Session Border Controllers. In *IEEE International Conference on Communications, ICC'06*, 11–15 June 2006, Istanbul, Turkey.
23. Schulzrinne, H., & Wedlund, E. (2000). Application-layer mobility using SIP. *Mobile Computing and Communication Review*, 4(3), 47–57.
24. Xlite SIP User Agent, <http://www.counterpath.com/>



Stefano Salsano was born in Rome in 1969. He received his honours degree in Electronic Engineering from the University of Rome (Tor Vergata) in 1994. In 1998 he was awarded a PhD from the University of Rome (La Sapienza). Between the end of 1997 and 2000, he worked with Co-RiTeL, a telecommu-

nications research institute, where he was co-ordinator of IP-related research. Since November 2000 he has been an assistant professor (“Ricercatore”) at the University of Rome (Tor Vergata) where he teaches the courses on “Telecommunications transport networks” (“Reti di trasporto”) and on “Telecommunication networks”. He has participated in the EU funded projects INSIGNIA (on the integration of the B-ISDN with Intelligent Networks), ELISA (on the integration of IP and ATM networks, leading the work package on Traffic Control), AQUILA (QoS support for IP networks, leading the work package on Traffic Control), FIFTH (on internet access via satellite on high-speed trains), SIMPLICITY (on simplification of user access to ICT technology, leading the work package on architecture definition), E2R (on reconfigurable radio systems), SMS (on service authoring platforms for mobile services, leading the work package on system architecture). He has also participated in ESA founded projects (ATMSAT, ROBMOD) and in national projects founded by the Italian Ministry of University and Research (MQOS, NEBULA, PRIMO, TWELVE). His current research interests include Ubiquitous Computing and Wireless LANs, IP telephony, Peer-to-peer networking architectures, QoS and Traffic Engineering in IP networks. He is co-author of more than 50 publications on international journals and conferences.



Luca Veltri received his Laurea degree in Telecommunication Engineering and his Ph.D. in Communication and Computer Science from University of Rome “La Sapienza” in 1994 and 1999, respectively. In 1999 he joined CoRiTeL, a research consortium founded by Ericsson Telecomunicazioni, where he worked until 2002. Since 2002 he is assistant professor at University of Parma (Italy), where is

currently working and where he teaches classes on Telecommunication Networks, Multimedia Communications, and Network Security. His main research fields are: IP telephony and P2P systems, mobility management in next generation networks, and network security. He worked in several research projects founded by the EU, by ESA and by the Italian Ministry of Research.



Alessandro Ordine received his “Laurea Specialistica” Degree in Telecommunications Engineering (University of Rome “Tor Vergata”) in April 2004 with a thesis on “Controlling TCP Fairness in WLAN access network”. Since November 2004, he is a PhD student at University of Rome “Tor Vergata”. He has participated in the EU project FIFTH (on internet access via satellite on high-speed trains) and in the Italian PRIN TWELVE (on the service differentiation in 802.11 networks). His current research interests include analysis of TCP and VoIP over Wireless LANs.

net access via satellite on high-speed trains) and in the Italian PRIN TWELVE (on the service differentiation in 802.11 networks). His current research interests include analysis of TCP and VoIP over Wireless LANs.



Andrea Polidoro received his “Laurea Specialistica” degree in Telecommunications Engineering (University of Rome “Tor Vergata”) in July 2005 with a thesis on “Reduction of signaling traffic in MPLS-TE network” (“Ingegneria del traffico in una rete MPLS-TE meccanismi per la riduzione del carico di segnalazione”).

Since November 2005, he is a PhD student at University of Rome “Tor Vergata”. Currently he is working on IP telephony and IP mobility research activities.