Seamless SIP-Based Mobility for Multimedia Applications

Nilanjan Banerjee, Motorola India Research Labs
Arup Acharya, IBM T. J. Watson Research Center
Sajal K. Das, The University of Texas at Arlington

Abstract

Application-level protocol abstraction is required to support seamless mobility in next-generation heterogeneous wireless networks. Session Initiation Protocol (SIP) provides the required abstraction for mobility support for multimedia applications in such networks. However, the handoff procedure with SIP suffers from undesirable delay and hence packet loss in some cases, which is detrimental to applications like voice over IP (VoIP) or streaming video that demand stringent quality of service (QoS) requirements. In this article we present a SIP-based architecture that supports soft handoff for IP-centric wireless networks. Soft handoff ensures that there is no packet loss and that the end-to-end delay jitter is kept under control.

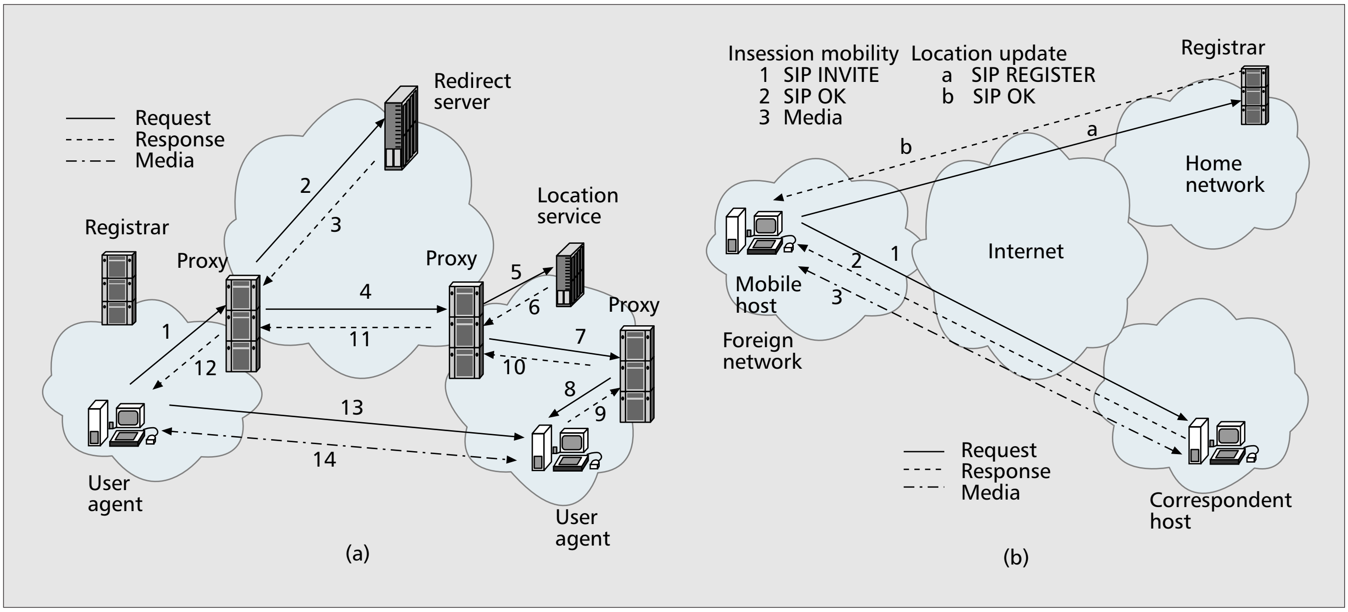
Seamless mobility in converged wireless Internet Protocol (IP)-centric networks provides an important paradigm for uninterrupted multimedia services. Seamless services require network and device independence that allow the users to move across different access networks and change computing devices. IP convergence has led to the coexistence of several IP-based wireless access technologies such as General Packet Radio Service (GPRS), CDMA 2000, and Wireless LAN, as well as the emergence of next-generation technologies like Universal Mobile Telecommunications System (UMTS). This coupled with the diverse range of multimodal mobile devices (e.g., Motorola CN620, Nokia 9500 Communicator) make seamless service provisioning extremely challenging, particularly for multimedia streaming applications (e.g., VoIP or video streaming) having stringent quality of service (QoS) requirements such as minimum bandwidth, delay, jitter, and loss rate.

Mobility management protocols are in general responsible for supporting seamless services across heterogeneous wireless access networks that require connection migration from one network to another. This is known as *vertical handoff*. Thus, in addition to providing location transparency, the mobility management protocols also need to provide network transparency. A number of protocols [1, 2] have been proposed for solving the vertical handoff problem for IP-based heterogeneous networks. Although these protocols have a common goal of location transparency, they differ from each other with regard to choices made during the design and implementation phases. They can be broadly classified based on the layer of their operation. For example, Mobile IP [1] works in the network layer, TCP-Migrate [2] in the transport layer, and Session Initiation Protocol (SIP) [3] in the application layer. The dependency of these mobility protocols on the access networks reduces progressively as we move up the protocol stack. For a comparative discussion and analysis, the reader can refer to [4]. Among these protocols, Mobile IP and SIP have been standardized by the Internet Engineering Task Force (IETF).

Mobile IP seems to be the architecturally right protocol for providing IP mobility, but it requires significant changes in the underlying networking infrastructure as well as the mobile hosts. Besides, Mobile IP suffers from the problem of triangular routing (which is detrimental to real-time traffic like streaming multimedia), where the important issues are fast handoff, low latency, and minimal packet loss. Although solutions exist for Mobile IP route optimization, the IP stack needs change in order to implement route optimization, which can only be initiated by the home agent. This introduces additional delay. Moreover, mobile IP encapsulation adds 8–20 bytes of overhead for each data packet.

TCP-Migrate also suffers from the drawback that it has to modify the TCP protocol implementation for all the hosts. Application-layer protocols, on the other hand, are transparent to lower-layer characteristics. For example, an application-layer protocol, sending user datagram protocol (UDP) packets, does not need to know how an underlying GPRS or a CDMA 2000 network transports the packet. The application-layer protocols maintain the true end-to-end semantics of a connection and are expected to be the right candidate for handling mobility in a heterogeneous network environment. SIP has been accepted by the Third-Generation Partnership Project (3GPP) as an application-layer signaling protocol for setting up real-time multimedia sessions. SIP is also capable of supporting terminal mobility as well as session mobility, personal mobility, and service mobility [5]. Recently, SIP has gained widespread acceptance from commercial vendors such as Sprint PCS and Verizon to provide important services such as Instant Messaging, push-to-talk, and so forth. Thus, SIP-based mobility management could potentially use a readily available operational infrastructure, which would facilitate its fast deployment. Therefore, SIP seems to be an attractive candidate as an application-layer mobility management protocol for vertical handoff [4]. Although the SIP-based mobility management scheme solves the problem posed by Mobile IP route opti-

0890-8044/06/$20.00 © 2006 IEEE



■ Figure 1. *SIP architecture and mobility support: a) SIP architecture; b) SIP-based mid-call terminal mobility management.*

mization, in some cases it introduces unacceptable handoff delays [6], particularly for multimedia applications with stringent QoS requirements. Furthermore, SIP entails application-layer processing of messages, which may introduce additional delay.

In this article, we present an architecture for an SIP-based mobility management supporting soft handoff scheme at the IP layer for next-generation wireless infrastructure networks. Soft handoff ensures minimal packet loss and handoff-delay variation, which are critical requirements for providing QoS to multimedia applications. The proposed architecture is inspired by existing works that introduce some level of data redundancy to solve the problem of packet loss. For example, a multicast-based architecture for host mobility [7] was proposed to reduce handoff delay and minimize packet loss. This approach needs the deployment of an IP multicast infrastructure. However, as IP multicast has not been that successful, this approach is subject to doubts regarding performance efficiency and deployment feasibility. Transport- and network-layer bandwidth aggregations [8, 9], in which multiple interfaces are used during handoff, were proposed to attain the same goal. An optimized handoff mechanism for SIP mobility, similar to the Mobile IP regional registration concept, was proposed in [10]. Soft handoff at the IP level for SIP-based mobility management was first suggested in [5]. A similar approach, based on CDMA's soft handoff mechanism, was proposed in [11] for optimized fast handoff schemes with SIP in CDMA networks. However, this study utilizes the multiple concurrently received signals in CDMA networks to achieve soft handoff. In contrast, in our proposed architecture, soft handoff is achieved at the IP layer with the help of SIP signaling, so that it is independent of the underlying radio access technology. We implemented this architecture in a testbed environment as a proof of concept, and evaluated its performance efficiency. The experimental results demonstrate that our architecture performs efficiently in terms of packet loss and delay jitter. A preliminary version of this article appeared in [12].

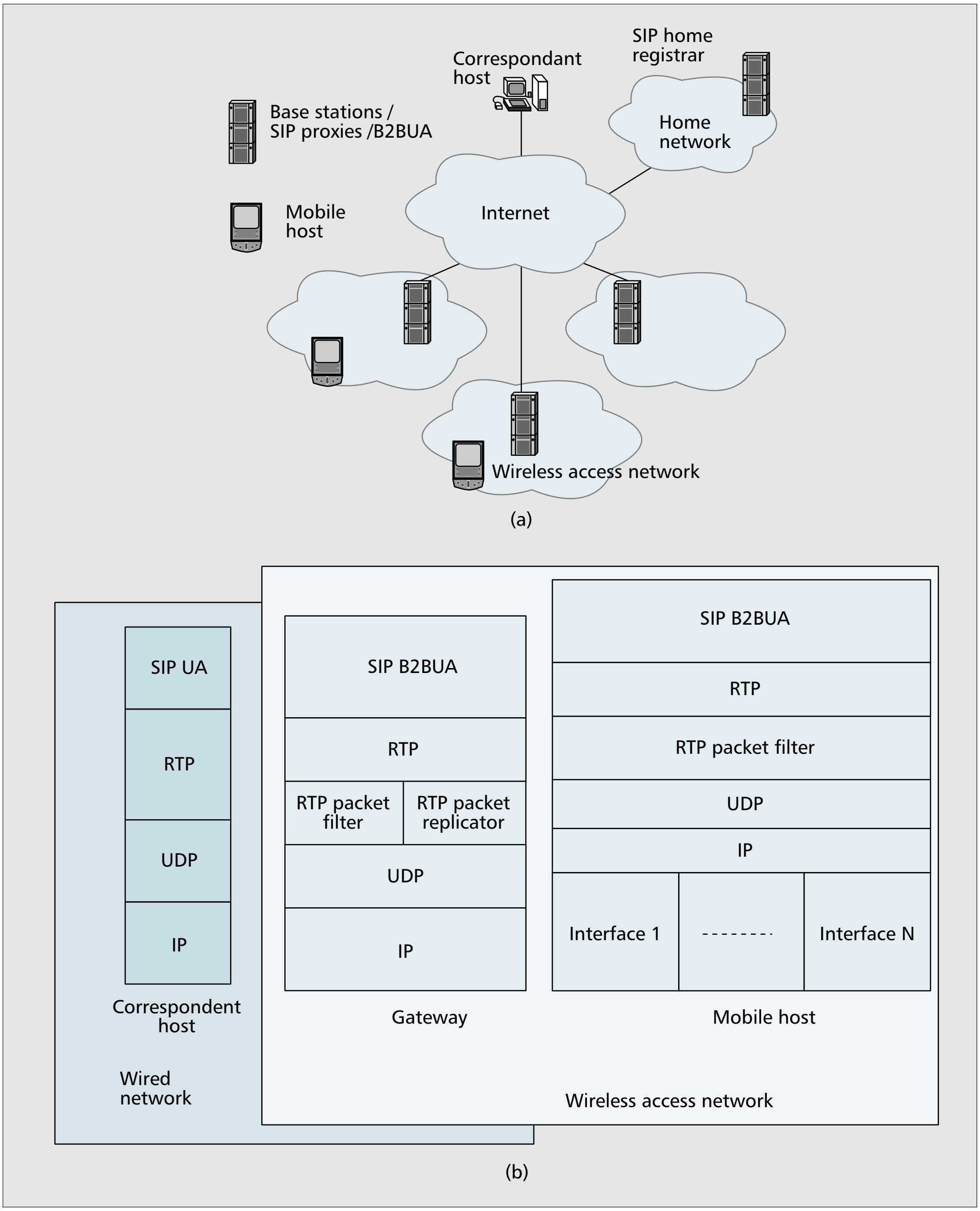
The rest of the article is organized as follows. We start with an overview of SIP and its mobility support, followed by a description of the problem with SIP-based vertical handoff and the proposed architectural solution. We then present an experimental study of our mobility architecture and discuss performance issues.

Overview of SIP and Mobility Support

SIP [3] is a signaling protocol for broadband multimedia applications that allows creation, modification, and termination of sessions with one or more participants. It is used for both voice and video calls, either for point-to-point or multiparty sessions. SIP is independent of the media transport, which for example, typically uses Real-Time Transport Protocol (RTP) over UDP. It allows multiple endpoints to establish and maintain media sessions between each other. This includes terminating the session, locating the endpoints, establishing the session, and modifying the media session after session establishment has been completed. Recently, SIP has gained widespread acceptance and deployment among wireline service providers for introducing new services such as VoIP, within the enterprises for Instant Messaging and collaboration, and among mobile carriers for push-to-talk services. Industry acceptance of SIP as the protocol of choice for converged communications over IP networks is thus highly likely.

As shown in Fig. 1a, a SIP infrastructure consists of user agents, registration servers, location servers, and SIP proxies deployed across a network. A user agent is a SIP endpoint that identifies services such as controlling session setup and media transfer. User agents are identified by SIP uniform resource identifiers (URIs) of the form `sip:user@domain`. All user agents register their IP addresses with a SIP registrar server that can be co-located with a SIP proxy. SIP defines a set of messages, such as INVITE, REGISTER, REFER, and so on, in order to set up sessions between the user agents. These messages are routed through SIP proxies that are deployed in the network. The Domain Name System (DNS) records help in finding SIP proxies responsible for routing the messages tothe destination domain.

A session or dialog is set up between two user agents following a client–server interaction model, where the requesting user agent client (UAC) interacts with the target user agent server (UAS). A logical entity formed by concatenating a UAC and a UAS is known as a back-to-back user agent (B2BUA), which keeps all the dialog information and intercepts all the participating messages within a dialog. All requests from an originating UAC (such as INVITE) are routed by the proxy to an appropriate target UAS, based on the target SIP URI included in the `Request-URI` field of the INVITE message header. Proxies may query location and



■ Figure 2. *a) The next-generation wireless network architecture; b) the proposed protocol architecture.*

redirect servers for SIP service discovery or to determine the current bindings of the SIP URI. Signaling messages are exchanged between user agents, proxies, and redirect/location servers to locate the appropriate services or endpoints for media exchange. For scalability, multiple proxies are used to distribute the signaling load. A session is setup between two user agents through SIP signaling messages comprising an INVITE (messages 1,2,4,7, and 8 in Fig. 1a), an OK response (messages 9–12 in Fig. 1a), and an ACK (message 13 in Fig. 1a) to the response [3]. The call setup is followed by media exchange using RTP. The session is torn down through an exchange of BYE and OK messages.

Apart from the session setup function, SIP inherently supports personal mobility and can be extended to support service and terminal mobility [5]. Personal mobility enables a user to be found, independent of the location and network device. Terminal mobility, on the other hand, enables a user to change location or IP address during the traffic flow of an ongoing session. It can be explained with an example of an ongoing session between a mobile host (MH) and a correspondent host (CH) as follows. Each MH belongs to a home network with a SIP server providing a registrar service. Each time the MH changes location, it registers with the home network's registrar service. In principle, this is similar to Mobile IP home registration. For ongoing sessions, the MH sends a re-INVITE message to the corresponding CH using the same call identifier as in the original setup. The former procedure takes care of *pre-call mobility*, while the latter enables *mid-call mobility*. High-level messaging of SIP-based mid-call mobility management is depicted in Fig. 1b. The new contact information (e.g., URI for future contact) is put in the Contact field of the SIP message so as to redirect the subsequent SIP mes-

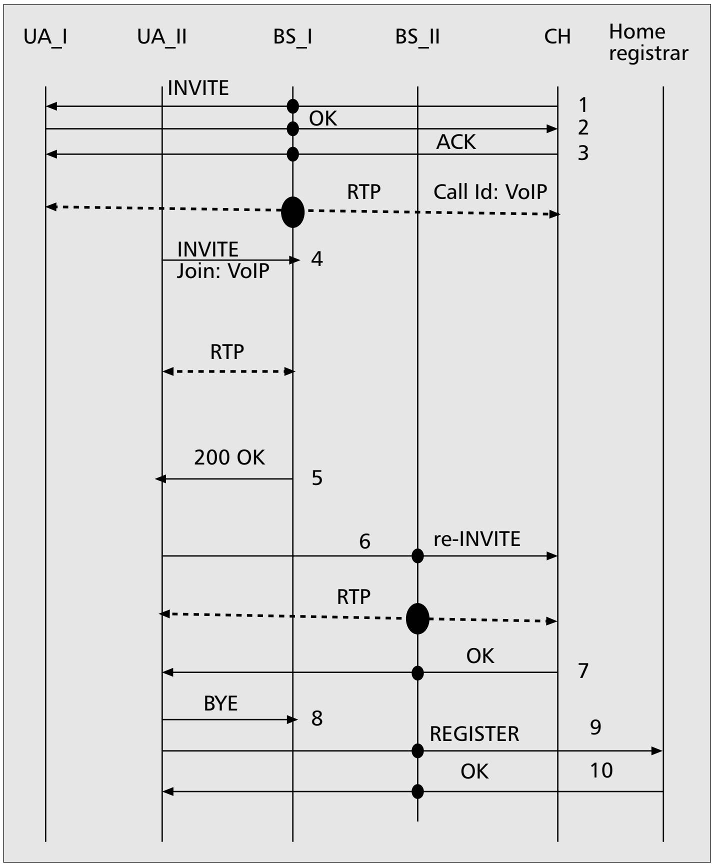


Figure 3. *Message diagram.*

sages to the current location. The data traffic flow is redirected by updating the transport address field in the Session Description Protocol (SDP) part of the re-INVITE message. For mid-call mobility, the CH starts sending data to the new location as soon as it gets the re-INVITE message. Hence, the handoff delay is essentially the one-way delay for sending an INVITE message from the MH to the CH. The problems with mid-call handoff delay are discussed below.

Problem with Mid-Call Handoff

Mid-call mobility is usually achieved by supporting handoff, the process of changing parameters (e.g., endpoint address, channel, etc.) associated with the current connection. For UDP-based connections, the major parameters are the source and destination IP addresses, which can be changed by the movement of an MH, either within one network (horizontal handoff) or across different networks (vertical handoff). Handoff can be *hard* or *soft* — characterized by "break before make" or "make before break" connections, respectively. In hard handoffs, current resources are released before new resources are used, whereas in soft handoffs both existing and new resources are used during the handoff process. For soft handoff, the MH should be capable of communicating through multiple network interfaces.

Usually, a mobility management protocol operating at the control plane (independent of the data plane) supports handoff. As mentioned above, SIP provides mid-call handoff support in IP-centric networks for multimedia applications. Although signaling protocols like Resource Reservation Protocol (RSVP) provide end-to-end QoS to the applications, it is the responsibility of the mobility management protocol to maintain QoS during the handoff period. For multimedia streaming applications, the most important QoS parameters are

- End-to-end delay
- Delay jitter or variation of end-to-end delay between the packets
- Packet loss

Of these, the first two parameters primarily depend on the network conditions in the path of the data traffic. Generally, the issues related to these parameters can be resolved by providing a playout and jitter buffer. The handoff delay causes only a glitch as far as these two parameters are concerned and has no long-term effect. However, large handoff delay causes considerable packet loss, which seriously affects the quality of the multimedia streaming applications. For example, approximately four to five voice packets are dropped with a handoff delay of 1 s for a 16 kb/s stream with 64 bytes voice packets; and 2×10^5 packets are lost for a 1.5 Mb/s MPEG-4 stream with 1050 bytes of packet size. Such packet dropping has serious consequences for the video quality because of the error propagation in MPEG-4, particularly with regard to the dependent frames or the I-frames [13]. For voice streams, packet loss usually results in annoying popping and clicking sounds.

The handoff delay in SIP-based mobility is essentially the time required by the re-INVITE message to reach the CH from the MH, but several different operations need to be completed before the INVITE message could be transported. These are:

- Detection of the new network by the MH. This depends on the networking technology (e.g., periodic beacons from the access points are used in wireless LANS (WLANs) to intimate a mobile device about the presence of the network) as well as on the operating system in the MH.
- The MH needs to acquire an IP address through a procedure specific to the access network. This may be a dynamic host configuration protocol (DHCP) address for WLANs or Attach and Packet Data Protocol (PDP) Context Activation for GPRS networks.

The analytical study in [6] revealed that the handoff delay can be more than 1 s for low-bandwidth access networks, for which hard handoff, according to the previous discussion, has a considerable effect on the application QoS. So, the mobility management protocol needs to employ some mechanism to counter the harmful effect of the handoff delay. Soft handoff technique provides such a mechanism to deal with the large handoff delays and consequent packet drop.

Proposed Architectural Solution

In this section, we describe the architecture for SIP-based mobility management supporting soft handoff at the IP layer in next-generation heterogeneous wireless networks. As illustrated in Fig. 2a, an MH can move between various wireless networks with different access technologies such as GPRS, CDMA, WLAN, and so forth. The MH is also equipped to interface with different types of access technologies and can receive/transmit packets through more than one of these interfaces simultaneously. In each wireless access network, the MH communicates through base stations acting as gateways to the Internet. The gateways function as outbound SIP proxies apart from providing services such as DHCP and SIP registrar service. Each MH is SIP-enabled and SIP takes responsibility for session setup and the provisioning of seamless mobility. According to the SIP architecture, each MH has a home network with a registrar service containing the latest location information of the MH. Typically, the CH that wants to setup

```
Message 1:  CH -> UA_I

INVITE sip:MH@home.com SIP/2.0
Via: SIP/2.0/UDP <IP address of CH>:5060
To: <MH@home.com>
From: <CH@correspondent.com>;tag=001
Call-Id: VoIP
CSeq 1 INVITE
Contact: <sip:CH@correspondent.com>
Record-Route: <sip:BS_I@visited_I.com;1r>

Message 2: UA_I -> CH

SIP/2.0 200 OK
To: <MH@home.com>;tag=002
From: <CH@correspondent.com>;tag=001
Call-ID: VoIP
CSeq 1 INVITE
Record-Route: <sip:BS_I@visited_I.com;1r>

Message 3: CH -> UA_I

ACK sip:MH@visited_II.com SIP/2.0
To: <MH@home.com>;tag=002
From: <CH@correspondent.com>;tag=001
Call-ID: VoIP
CSeq 1 INVITE

Message 4: UA_II -> BS_I

INVITE sip: BS_I@visited_I.com SIP/2.0
Via: SIP/2.0/UDP <IP address of UA_II>:5060
To: <sip:BS_I@visited_I.com>
From: <sip:MH@home.com>;tag=003
Call-Id: VoIP
CSeq: 2 INVITE
Contact: <MH@visited_II.com>
Join: VoIP;to-tag=001; from-tag=002

Message 5: BS_I -> UA_II

SIP/2.0 200 OK
To: <sip:BS_I@visited_I.com>;tag=004
From: <sip:MH@home.com>; tag=003
Call-ID: VoIP
CSeq 2 INVITE

Message 6: UA_II -> CH

INVITE sip: CH@correspondent.com SIP/2.0
Via: SIP/2.0/UDP <IP address of UA_II>:5060
To: <sip:MH@home.com>;tag=002
From: <sip:CH@correspondent.com>;tag=001
Call-ID: VoIP
CSeq: 3 INVITE
Contact: <sip:MH@visited_II.com>

Message 7: CH -> UA_II

SIP/2.0 200 OK
To: <sip:MH@home.com>;tag=002
From: <sip:CH@correspondent.com>;tag=001
Call-Id: VoIP
CSeq 3 INVITE

Message 8: UA_II -> BS_I

BYE sip: BS_I@visited_I.com SIP/2.0
Via: SIP/2.0/UDP <IP address of UA_II>:5060
To: <sip:MH@home.com>;tag=002
From: <sip:CH@correspondent.com>;tag=001
Call-ID: VoIP
CSeq: 1 BYE

Message 9: UA_II -> Home Registrar

REGISTER sip: registrar.home.com SIP/2.0
Via: SIP/2.0/UDP <IP address of UA_II>:5060
To: <sip:registrar@home.com>
From: <sip:MH@home.com>;tag=005
Call-ID: abcd
CSeq: 1 REGISTER
Contact: <sip:MH@visited_II.com>

Message 10: Home Registrar -> UA_II

SIP/2.0 200 OK
To: <sip:registrar@home.com>;tag=006
From: <sip:MH@home.com>;tag=005
Call-ID: abcd
CSeq: 1 REGISTER
Contact: <sip:MH@visited_II.com>
```

Figure 4. *Message description.*

a session with the MH contacts the registrar service at the MH's home network and gets the latest contact information for the MH. As described above, when an MH moves to a different network acquiring a new IP address, its SIP client initiates a handoff procedure by sending a re-INVITE message with updated SDP parameters to the CH as well as to the home network's registrar service. Handoff can also be base-station-assisted, but we have adopted an MH-initiated handoff, as it has the best knowledge of the currently active network interfaces and hence is the best candidate to initiate the handoff.

SIP-Based Soft Handoff

The soft handoff procedure is initiated by the MH but executed at the base stations. The corresponding protocol architecture is shown in Fig. 2b. Each base station is equipped with a SIP B2BUA and a SIP proxy server. A B2BUA is a logical entity that receives a request and processes it as a UAS. It maintains dialog state and participates in all requests sent on the dialog it has established. All the SIP messages are directed through the outbound proxy at the base station using the `Record-Route` field of the message header, so that the B2BUA is able to capture the ongoing dialog information. The B2BUA is coupled with a *media gateway* that acts as a proxy, forwarding the RTP packets. The media gateway has dual functionality as an RTP *packet replicator* and an RTP *packet filter*. The replicators and filters are configured by the B2BUAs so that they act only on the desired media streams requiring soft handoff support. (The interaction between the media gateway and the B2BUA is further explained below.) The MH, on the other hand, has a packet filter only.[1] The packet replicator duplicates an RTP packet and sends it to a different IP address, while the packet filter eliminates duplicate RTP packets received at the media gateway and sends a single copy of the RTP packet to the destination. In principle, the B2BUA agent and the media gateway can be physically decoupled from each other.

When an MH is in transition from one network to another (i.e., during the handoff period), more than one network interface become active and the MH is capable of communicating

[1] The proposed architecture, as shown in Fig. 2b, has been designed primarily for downstream traffic from the CH. The MH would typically need an RTP replicator for implementing soft handoff for upstream traffic as well.

through them. Now, SIP does not enforce any restriction on the use of the network interface while sending SIP messages. In fact, any of the available network interfaces can be used by a SIP user agent to send the messages and this facility is available in almost all of the SIP client implementations. During the transition period when a new network interface becomes activated, the SIP UAC at the MH sends an INVITE message with the JOIN header [14] to the SIP B2BUA proxy server. Note that for this operation, the SIP client only requires knowledge about the available network interfaces during the handoff period and requires no other support from the network layer. Thus, although the soft handoff takes place at the IP layer, it is entirely controlled at the application layer. The JOIN header contains all the relevant information about the ongoing call. The B2BUA, being a stateful entity, is able to identify the call and accordingly configures the packet replicator and the packet filter. The B2BUA essentially configures the packet replicator at the media gateway to send a copy of all packets directed towards the old interface of the MH to the newly activated interface. During the transient handoff period, the MH sends and receives the packets through both the interfaces. The packet filters at the media gateway and the MH discards the duplicate RTP packets. As soon as the packet reaches the MH through the newly activated interface, a re-INVITE message is sent to the CH with the IP address for the newly active interface and the corresponding contact information. As a result, the call parameters are renegotiated on an end-to-end basis, with the selection of a new intermediate SIP proxy server and B2BUA belonging to the base station corresponding to the newly activated interface. Once the call renegotiation is complete, a BYE message is sent to terminate the call-leg through the old interface, as soon as a duplicate packet reaches the newly activated interface. Finally, the MH registers its new location information with the home network's registrar service by using REGISTER message. The concept of SIP-based soft handoff is further illustrated by the following example.

An Example

Let us assume that a session is in progress between the CH and the MH, which belong to different subnet domains with the SIP URIs `CH@correspondent.com` and `MH@home.com`, respectively. The MH moves between two domains, namely, `visited_I.com` and `visited_II.com`. The corresponding base stations for the two domains are denoted as BS_I and BS_II with URIs as `BS_I@visited_I.com` and `BS_II@visited_II.com`, respectively. The MH has two interfaces, namely, UA_I and UA_II, through which it acquires IP address pertaining to the two domains.

When the MH moves from domain `visited_I.com` to `visited_II.com`, UA_II becomes activated and acquires an IP address through a mechanism specific to that particular network. The MH SIP UA, on detecting the newly activated UA_II interface, then sends an INVITE message, with a JOIN header option, to BS_I through interface UA_II. The INVITE message has the new contact address `MH@visited_II.com` for the MH in the `Contact` field. The SDP parameters are also updated with the newly acquired IP address. The JOIN header contains information (call-id, to-tag, and from-tag), which is used by B2BUA at BS_I to match the existing SIP dialog corresponding to the media session in consideration. BS_I then configures the RTP packet replicator and the filter for the particular ongoing dialog in order to send a copy of packets directed toward UA_I to UA_II and filter duplicate packets coming from the MH via the two interfaces. At the same time, a SIP OK message is sent to UA_II. Therefore, for a transient period, the RTP packets reach both interfaces of the MH. The duplicate RTP packets at the MH are filtered by the packet filter and delivered to the upper layers, while those at the media gateway are filtered and sent to the CH.

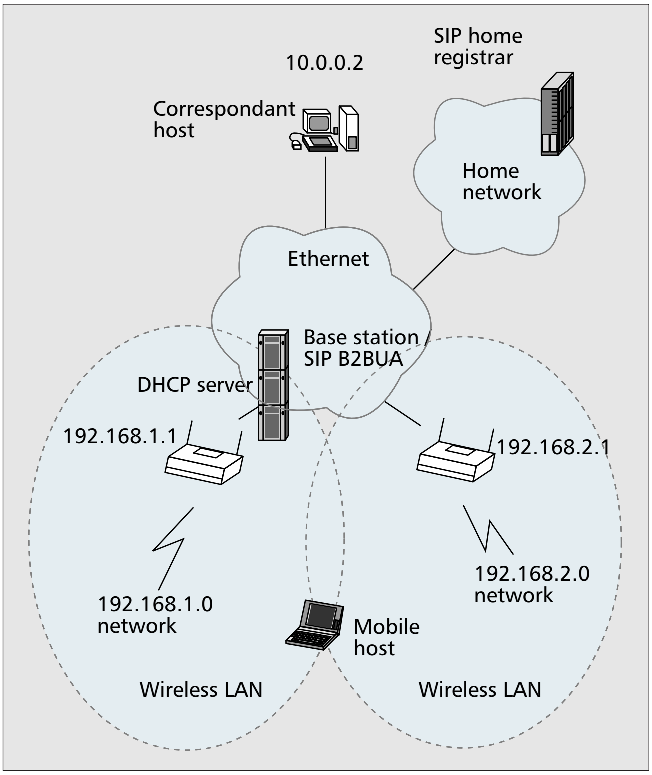


Figure 5. *Experimental testbed setup.*

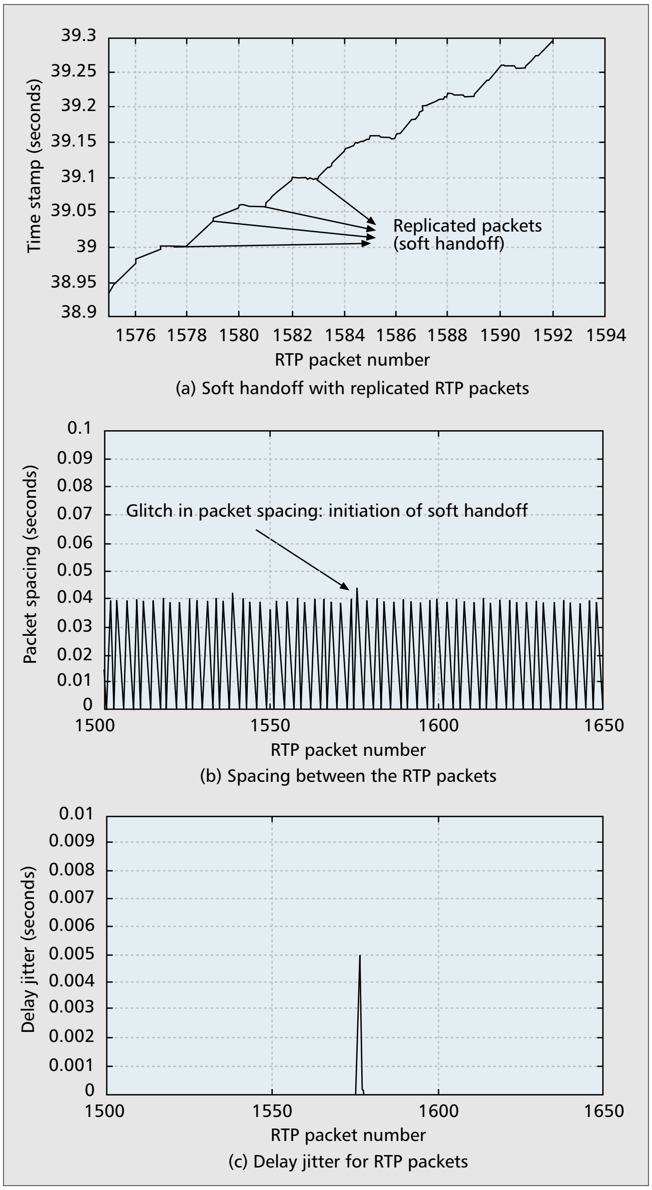
As soon as the MH starts receiving the packets through UA_II, it sends a re-INVITE message to the CH to renegotiate the session parameters on an end-to-end basis, with changed endpoints. As a result of session renegotiation, the path of the media packets gets straightened out and the CH communicates with the MH through BS_II. As soon as a duplicate packet reaches the interface UA_II, the connection from UA_I is released by sending a BYE message to BS_I, so that it can delete the dialog information pertaining to the SIP dialog going through BS_I. The timing diagram for the example is depicted in Fig. 3 and the detailed description of each of the messages is given in Fig. 4 (only the headers are shown due to lack of space).

The handoff procedure is composed of the following major operations, each of which contributes to the handoff delay:
- Network detection and address configuration operation performed by the MH. This depends on the networking technology and the MH's operating system.
- Sending the INVITE message with the JOIN header to BS_I.
- Sending the re-INVITE message to update the session with the new location parameters.

The corresponding delays are denoted by t_{attach}, t_{join}, and $t_{re\text{-}invite}$, respectively. As mentioned above, these delays cause considerable packet loss, which aversely affects the QoS of multimedia streaming applications. The objective of the proposed architecture is to nullify the effect of these delay components with soft handoff.

Experimental Setup

The proposed mobility architecture has been implemented in an experimental testbed, shown in Fig. 5, for performance evaluation. Two different IEEE 802.11b wireless LAN-based subnets were created. `Linphone 0.12.2`, a



■ Figure 6. *Performance results.*

Gnome-based SIP soft phone built on a GNU `oSIP2 2.0.6` stack is used as the SIP UAC and UAS for session management. Linphone allows the selection of network interface for SIP message communication. The SIP B2BUA at the base station is implemented by modifying `siproxd 0.5.4`, a stateful SIP proxy, also built on the `oSIP2` stack. The proxy has been modified to understand the semantics of the JOIN message in the context of soft handoff and to activate the packet replication rules. Note that the replication and filtering functionalities can be implemented using RTP translators [15]. However, RTP translators are logical units and depend on the implementation of the RTP stack. But, since packet replication at the RTP layer affects performance considerably, we have implemented the replication functionality at the IP layer using user-space tools such as `iptables`, available in the linux 2.4 kernel. Simple `iptables` rules duplicating UDP packets based on the destination IP address and RTP port number are set at the gateway through SIP messaging. Filtering, however, is done using the RTP translator approach which filters duplicate RTP packets based on an RTP synchronization source (SSRC) identifier at the RTP level.

Performance Gains

The performance of the proposed soft handoff architecture was measured in the testbed described above using a captured voice stream coded with Speex 8000 codec. Typical observed values of the parameters are as follows: t_{attach} = 23.95369231 secs, t_{join} = 3.618 msecs, and $t_{re\text{-}invite}$ = 359.84 msecs. Note that the measurement for t_{attach} is considerably high for the following reason. We have considered a handoff scenario in which the network detection and configuration process starts when the MH receives a weak signal from the new base station. Due to inadequate signal strength during the transition period, the configuration process takes considerable time because of repeated transmission failures for DHCP messages. To demonstrate the effect of soft handoff, the MH was moved into a new subnet after 15 s. Due to the handoff delay components, the soft handoff procedure could not be initiated before 38.9572 s. The soft handoff initiation points are indicated in Fig. 6a. The vertical notches in the plot imply duplicated RTP packets received at the MH that are subsequently filtered out by the packet filter. The packet replication continues until the re-INVITE message updates the session parameter, which enables the CH to redirect the packets directly to the MH at its new IP address. As expected, no packet loss was observed in the RTP stream.

Figure 6b shows the spacing in seconds between the consecutive RTP packets. For the purpose of clarity, a portion of the stream (from packet 1500 to 1650 only) is shown with a glitch in the interpacket spacing, which indicates the point at which the MH stops accepting packets through the old interface and starts accepting them through the newly activated interface. The glitch results from the different routes taken by the packets directed toward the old and the new interfaces. The delay jitter is typically a measure of the difference in the end-to-end delay along the two different routes corresponding to the two network interfaces and is shown as the single spike in Fig. 6c. However, other than these glitches, the jitter remains under control all the time and has no long-term effect on the streaming RTP traffic. The standard deviation of the observed packet spacings is 17.125 ms and that for the delay jitter is only 0.112 ms. As mentioned above, such spikes in delay jitter can be nullified by using a playout and jitter buffer at the terminal device, without any support such as soft handoff from the network infrastructure.

Based on the above results, we conclude that the proposed architecture ensures zero packet loss and controlled delay jitter during vertical handoff between heterogeneous wireless networks.

Conclusion

SIP provides elegant application-layer mobility support that solves the problems associated with lower-layer mobility protocols in next-generation heterogeneous wireless access networks. However, the handoff delay in SIP may be substantial, thus causing considerable packet loss, which seriously affects the quality of voice or video streams. In order to alleviate the problem of packet loss, in this article we have presented a SIP-based mobility architecture for soft handoff in next-generation wireless networks. A testbed has been set up to measure the efficiency of the proposed architecture. The experimental results show that the architecture is capable of ensuring zero packet loss and controlled delay jitter.

Acknowledgments

This work is partially supported by the PSI project (NSF ITR grant no. IIS-0326505) at UT Arlington and ORBIT project (NSF NRT grant no. ANI-0335244) at Rutgers University.

References

[1] C. E. Perkins, "IP Mobility Support for IPv4," RFC 3220, Jan. 2002.

[2] A. C. Snoeren and H. Balakrishnan, "An End-to-End Approach to Host Mobility," *Proc. ACM Mobicom*, Aug. 2000, pp.155–66.

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

[4] N. Banerjee *et al.*, "Mobility Support in Wireless Internet," *IEEE Wireless Commun.*, vol. 10, no. 5, 2003, pp. 54–61.

[5] H. Schulzrinne and E. Wedlund, "Application-Layer Mobility Using SIP," *Mobile Comp. and Commun. Rev.*, vol. 4, no. 3, 2000, pp. 47–57.

[6] W. Wu *et al.*, "SIP-Based Vertical Handoff Between WWAN and WLAN," *IEEE Wireless Commun.*, Special Issue: Toward Seamless Internetworking of Wireless LAN and Cellular Networks, vol. 12, issue 3, June 2005, pp. 66–72.

[7] J. Mysore and V. Bharghavan, "A New Multicasting-Based Architecture for Internet Host Mobility," *Mobile Comp. and Net.*, 1997, pp. 161–72.

[8] K. Chebrolu and R. Ramesh, "Communication Using Multiple Wireless Interfaces," *Proc. IEEE Wireless Commun. and Networking Conf.*, vol. 1, 2002, pp. 327–31.

[9] H.-Y. Hsieh and R. Sivakumar, "A Transport-Layer Approach for Achieving Aggregate Bandwidths on Multihomed Mobile Hosts," *Proc. ACM Mobicom*, 2002, pp. 83–94.

[10] A. Dutta *et al.*, "Fast-Handoff Schemes for Application-Layer Mobility Management," *IEEE Int'l. Symp. Pers., Indoor and Mobile Radio Commun.*, vol. 3, 2004, pp. 1527–32.

[11] F. Vakil *et al.*, "Virtual Soft Hand-Off in IP-Centric Wireless CDMA Networks," *Proc. Int'l. Conf. 3G Wireless and Beyond*, 2001; http://www.argreenhouse.com/SIP-mobile/sipvirtual.pdf

[12] N. Banerjee, A. Acharya, and S. K. Das, "SIP-Based Mobility Architecture for Next-Generation Wireless Networks," *Proc. PerCom*, 2005, pp. 181–90.

[13] N. Feamster and H. Balakrishnan, "Packet Loss Recovery for Streaming Video," *Packet Video Wksp.*, Apr. 2002.

[14] R. Mahy and D. Petrie, "The Session Inititation Protocol (SIP) 'Join' Header," Feb. 2004, draft-ietf-sip-join-03.txt, work in progress.

[15] H. Schulzrinne *et al.*, "RTP: A Transport Protocol for Real-Time Applications," IETF RFC 1889, Jan. 1996.

Biographies

NILANJAN BANERJEE (nilanjan@motorola.com) is a senior research engineer in the Motorola India Research Laboratory. He received his Ph.D. and M.S. degrees in computer science and engineering from the University of Texas at Arlington. He received his B.E. degree in the same discipline from Jadavpur University, India. His research interests include telecom network architectures and protocols, identity management and network security, mobile and pervasive computing, measures for performance, modeling and simulation, and optimization in dynamic systems.

ARUP ACHARYA (arup@us.ibm.com) received his B.Tech degree in computer science and engineering from the Indian Institute of Technology, Kharagpur, and a Ph.D. in computer science from Rutgers University in 1995. He is a research staff member in the Internet Infrastructure and Computing Utilities department at IBM T. J. Watson Research Center and leads the Advanced Networking micropractice for On-Demand Innovation Services within IBM Research. His current research focuses on scalability and server enhancements for SIP and SIP-based applications such as VoIP, instant messaging, and presence, as well as cooperative protocols for wireless mesh networks. He has published in leading conferences and journals in the area of networking architecture and protocols, holds seven patents, and has chaired several conferences. He has been invited to present tutorials on SIP as well as mobile wireless networks at multiple international conferences. He was with NEC C&C Research Labs prior to joining IBM Research. Further information is available at http://www.research.ibm.com/people/a/arup/

SAJAL K. DAS (das@cse.uta.edu) is a professor of computer science and engineering, and also founding director of the Center for Research in Wireless Mobility and Networking (CReWMaN) at the University of Texas at Arlington (UTA). His research interests include mobile wireless communications, resource and mobility management in wireless networks, mobile and pervasive computing, wireless multimedia, ad hoc and sensor networks, mobile internet architectures and protocols, security, distributed and grid computing, and performance modeling and simulation. He has published more than 350 research papers in these areas in journals and leading international conferences, directed numerous industry and government funded projects, and holds five U.S. patents in wireless mobile networks. He received Best Paper Awards at ACM MobiCom '99, ICOIN '01, ACM MSWiM 2000, and ACM/IEEE PADS '97. He is a coauthor of the book *Smart Environments: Technology, Protocols and Applications* (Wiley, 2005). He serves as Editor-in-Chief of *Pervasive and Mobile Computing* and as an Associate Editor of *IEEE Transactions on Mobile Computing*, *IEEE Transactions on Parallel and Distributed Systems*, and *ACM/Springer Wireless Networks*. He has served as General and Program Chair and TPC member of numerous IEEE and ACM conferences. Further information is available at http://crewman.uta.edu.