Evaluation of VoIP in a Mobile Environment using an end-to-end Handoff Mechanism

John Fitzpatrick, Seán Murphy, Mohammed Atiquzzaman*, John Murphy

Performance Engineering Lab,
School of Computer Science and Informatics,
University College Dublin,
Ireland
E-mail: john.fitzpatrick@ucd.ie

**School of Computer Science,*
University of Oklahoma,
Norman, OK.
E-mail: atiq@ou.edu

Abstract **- In parallel with the growth of VoIP services, a large number of manufacturers have begun to develop multimode devices capable of connecting to multiple wireless access networks simultaneously. These multimode terminals have, in principle, the capability to support seamless handover between different radio access networks. One handover solution that utilises the IP diversity which these multimode devices can deliver is SIGMA. SIGMA is a promising end-to-end transport layer handover solution based on SCTP. The objective of this paper is to investigate if SIGMA is capable of providing seamless handover of VoIP calls without degradation in voice quality. To achieve this, we developed a Linux based testbed on which we implemented a VoIP client and server using SIGMA handoff. Each voice call uses the G.711 voice codec over RTP. PR-SCTP is used in place of the traditional UDP as the transport layer protocol. SIGMA based handover was used during full duplex voice calls between the client and server. The ITU-T E-Model has been used to calculate the voice quality during the handover. Results show that SIGMA can be used as a seamless handover mechanism for VoIP without any impact on voice quality.**

I. INTRODUCTION

The use and popularity of VoIP has been mainly confined to the wired domain or static wireless devices. However, with the growing availability and popularity of multimode capable devices, the ability to provide seamless handover mechanisms across multiple access technologies will become increasingly important. Critical to any handover mechanism for real time applications such as VoIP, is the ability to perform handovers that do not degrade the QoS of ongoing connections. Typically for high quality VoIP there must be no more than 1% packet loss and packet delay must not exceed 150ms. Therefore, for any handover mechanism to be considered seamless it must meet these minimum thresholds.

Vital to the success of any new network technology is ease of rollout. Any handover solution which requires the addition of new network nodes and upgrading of a large number of existing nodes will be stifled by slow deployment. A typical example of this is Mobile IP (MIP) [1], as discussed in [2], MIP has been around for 10 years, though during this time has seen very little rollout. The main problem is that for MIP to be viable it requires MIP capable nodes to be available in every network that a mobile host may visit. Since very few networks are MIP capable there is no motivation for businesses and providers to implement it. For this reason we argue that terminal oriented solutions that require no network infrastructure modifications provide a simpler solution in terms of deployment, requiring only software upgrades at the end terminals.

This paper is structured as follows. Section II discusses related work. Section III gives an overview of the SIGMA handover process. Section IV describes the testbed architecture and experimental setup. This is followed by a discussion of the results obtained in Section V. The paper is then concluded in Section VI.

II. RELATED WORK

By far the most popular IP mobility solution is MIP. Although MIP was not designed as a mobility solution for real time data such as VoIP, some work has been done in analysing MIP in this context.

In [12], MIP and SIP were analysed as candidates for VoIP mobility management. This work showed that in the majority of cases MIP resulted in smaller disruption times than SIP; however the disruptions were of the order of a few hundred milliseconds. This work used IPv4; however similar studies have been done using the mobility aspects of IPv6, which can reduce these delays.

In [3], analytical models were used to evaluate the performance of MIPv4 and MIPv6 in terms of handover latency for VoIP. The handover latency was defined as the time between the MN sending an agent solicitation message to the moment it could receive packets. The results showed the effect of increases in the wireless link delay on the handover latency of both MIPv4 and MIPv6. MIPv6 was found to be most affected as it involves the highest number of message exchanges over the wireless link. These results show delays of the order of a few hundred milliseconds as being common, making it unsuitable for supporting VoIP.

An analysis of the handoff delay for SIP over IPv6 was performed in [4]. This work showed SIP handover delays of between 2s and 40s, based on the IPv6 implementation for Linux. However, with kernel modifications the authors were able to reduce the delay to 450ms. Even with this improvement the delay is still too large to provide seamless handovers for VoIP.

Hierarchical MIP [5] improves the performance and scalability of MIP by introducing a two level hierarchy, allowing for both macro mobility and micro mobility. Although this solution greatly improves the handoff performance of MIP, it requires even greater infrastructure modifications with the addition of a mobility server at each level of the hierarchy.

Fast handovers for MIPv6 [6] utilises predictive information about the mobile nodes (MN) next point of attachment in order to realise a make before break solution. The MN obtains an IP from the new access router before breaking the connection with the current access router. In [7], the performance of FMIPv6 was analysed using ns2 with respect to handoff latency and packet loss rate. In this study it was shown that FMIPv6 can achieve delay values under 150ms with loss rates of less than 1%. Although this solution meets the delay and loss requirements of VoIP it still requires infrastructure modifications associated with traditional MIP.

Previous work has been performed on analysing the appropriateness of various mobility schemes for supporting real time applications such as VoIP. However, they focused on handoff latency and packet loss during handover and did not analyse how these impairments mapped to perceived voice quality. In this paper we obtain loss, jitter and delay measurements for SIGMA handoffs and evaluate their impact on perceived voice quality using the ITU-T recommended E-model.

III. SIGMA Overview

SIGMA is a transport layer handover scheme, which utilises IP diversity to perform seamless handovers between wireless networks. The MN uses two wireless interfaces – one for communicating with its current network and the other for making a connection to a new network, when one becomes available. Two different IP addresses obtained from the two different networks are bound to the interfaces. Handover to the new network is initiated based on a handover trigger which depends on the relative signal quality of the communications with the two interfaces. In this paper the multi-homing feature of SCTP is used to provide the IP diversity required by SIGMA. The multi-homing feature of SCTP allows a single association to span multiple IP interfaces regardless of the underlying network technology.

When an SCTP association is initiated between two endpoints, one of the IP interfaces at each endpoint is defined as the primary address. All communication to that endpoint is routed to its primary address until a failure is detected or an upper layer specifically requests the use of an alternate IP interface.

Addresses can be dynamically added and deleted from the association using SCTP's Dynamic Address Reconfiguration (DAR) extension [8]. The DAR extension to SCTP defines a new message type called an Address Configuration Message (ASCONF). This message can be transmitted by either endpoint to inform its peer of IP addresses through which it is reachable. This can be done dynamically during an active association and is the main feature that enables SCTP to support seamless handovers.

A. *SIGMA Handover Mechanism*

This section gives an overview of the SIGMA handover process. Figure 1 shows a timing diagram of the signalling involved in a SIGMA handover event. The SIGMA handover process can be defined in the following 4 steps.

1. Acquire new IP – When the MH moves into the coverage area of a wireless access network it is assumed that it can detect the availability of this network. For example in the experimental testbed which will be described later, 802.11 wireless access points were used. Hence, network detection can be done via the APs beacon frame router advertisement. On detection of the AP, it must authenticate, associate and obtain a new IP using DHCP, DHCPv6 or IPv6 Stateless Address Auto-Configuration (SAA).
2. Once the MH has acquired a new IP address it must add this to the association by informing the CN of the new IP. This is done using DAR.
3. In our implementation, the handover decision is based on the Received Signal Strength (RSS) of each available AP. When the RSS of the newly available AP becomes greater than that of the existing AP, a handover is triggered. To perform the handover SIGMA must redirect the data flow to the CN via the new AP. The handover is done by sending a '*Set Primary*' ASCONF message to the CN containing the new Primary address.
4. The final step in the handover process is to remove the old IP address from the association, so that no data is transmitted to the MH via the old access network which may no longer be available. Once again, an ASCONF message is transmitted to the CN containing the '*Delete IP*' parameter.

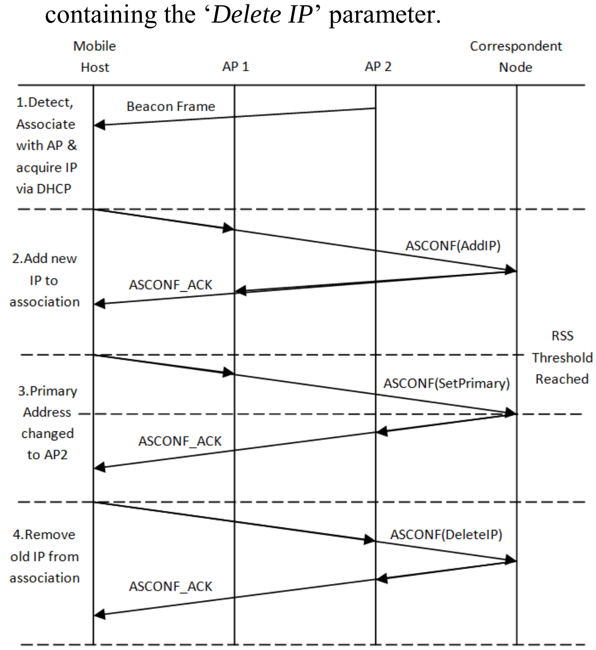

Figure 1 - SIGMA Signalling Diagram

IV. Testbed Architecture

In this section, we outline the network topology and architecture of the SIGMA VoIP testbed, as is shown in Figure

2. The testbed consists of an 802.11b AP, an 802.11g AP, two PCs to act as gateways between each AP, and an Ethernet LAN. Each AP uses DHCP to assign IP addresses to the Mobile Host (MH). The Mobile Host and the Correspondent Node (CN) were built on laptop computers running fedora core 5. The MH has two 802.11 WLAN cards to enable connection to both available WLANs simultaneously. The two WLAN cards of the MH give the IP diversity required by the SIGMA handover mechanism. The CN is connected directly to the LAN using Ethernet. The Linux Kernel implementation of SCTP (LKSCTP) was installed on each machine with both the ADD IP and Partial Reliability extensions enabled.

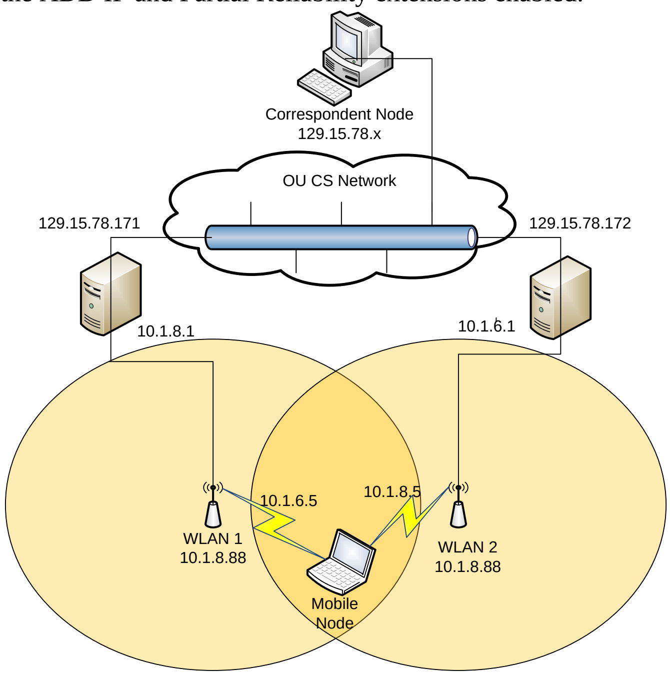

Figure 2 - Testbed Architecture

SCTP requires strict message ordered delivery and suffers from head of line blocking on a per stream basis. Although SCTP can support unordered message delivery the head of line blocking problem still exists. To combat this, we used the Partial Reliability extension to SCTP [9] to provide varying levels of reliability to upper layer protocols.

Partial Reliability uses 'Timed Reliability' which allows the sender to specify a time to live parameter on a per message basis. The time to live parameter defines the duration for which the sender should attempt to transmit and retransmit the message. If transmission is not successful within the specified timeframe, the message is dropped and a FORWARD_TSN message is transmitted to the receiver containing a new cumulative Transmission Sequence Number (TSN). On reception of the FORWARD_TSN message the receiver updates the TSN so that any missing messages up to the specified TSN will not be reported as missing in future SACKs. Essentially, PR-SCTP allows the sender to define how persistent the transport layer will be at attempting to deliver each message. The partial reliability extension to SCTP is not used by default; an SCTP endpoint can use partial reliability only if it is supported by its peer. An endpoint is notified that partial reliability is supported by its peer during association establishment.

A Client/Server application that uses PR-SCTP and emulates Constant Bit Rate (CBR) full duplex VoIP data was developed. Each full duplex stream comprised of two simplex streams, one for both the uplink and downlink. The motivation behind using voice calls as full duplex was based on the most commonly used VoIP program Skype, which uses full duplex (i.e. no silence suppression is used). Skype uses full duplex for two reasons. Transmitting silent packets maintains UDP bindings at NAT (Network Address Translation). Also, if data is being transmitted over TCP, the silent period packets prevent a reduction in the congestion size during the silent period.

The G.711 codec with a default frame size of 10ms was emulated in the client as the VoIP data source .The client encapsulates dummy VoIP data in RTP packets and transmits each packet to the server using PR-SCTP. On reception of each VoIP packet the server immediately echoes the packet back to the client. This creates a two way flow of VoIP data effectively emulating a full duplex VoIP call.

Each RTP header contains the packet transmission time. This enables the MH on receipt of the echo from the CN, to calculate the Round Trip Time (RTT) for each VoIP packet.

In this testbed setup, as described earlier, the client application runs on the MH and the server application on the CN. The system stack for both the client and server is shown in Figure 3.

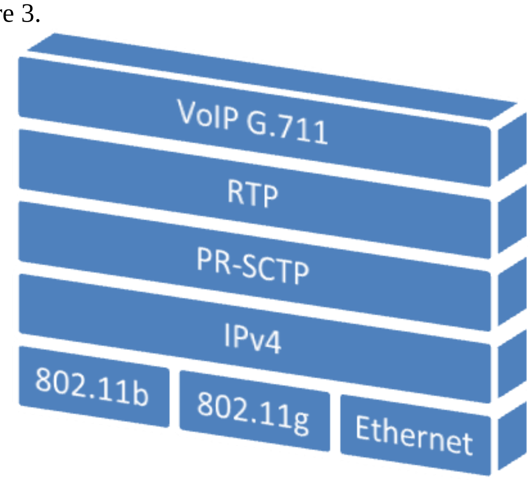

Figure 3 - Client/Server Stack

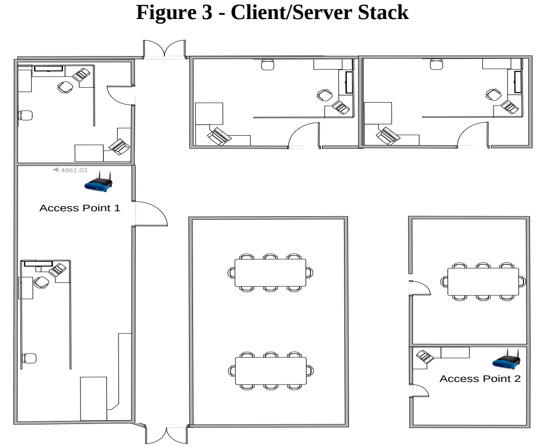

Figure 4 - AP placement in building

V. RESULTS

Two APs were used as described in the previous section, one at each end of a corridor as shown in Figure 4.

A call was set up between the MH and CN via the initial primary through AP1. The MH then moved toward the opposite end of the corridor in the direction of AP2. As can be seen in Figure 5, a handover is triggered when the RSS of AP2 becomes greater than that of AP1 plus a small hysteresis value of 10dbm. The 10dbm hysteresis value was introduced to prevent the ping pong effect of rapid handovers due to small network fluctuations.

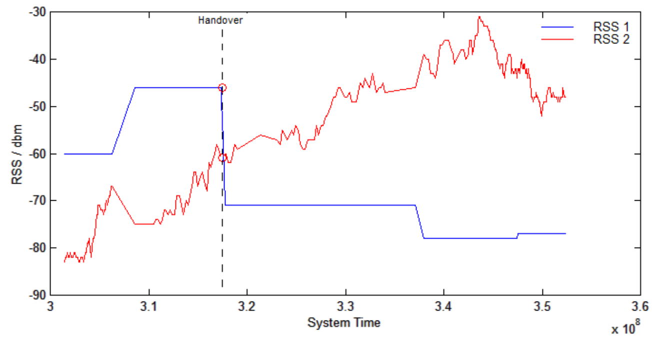

Figure 5 - RSS Values during Handover

The developed Client/Server VoIP application maintains a list of RTT values for each VoIP packet. This allowed the end-to-end delay and packet loss rate to be obtained. Also, based on the RTT values, a moving average of the packet jitter was calculated using the RTP jitter algorithm. The values do not give an insightful representation of how these impairments affect the quality of a VoIP call. To obtain a realistic depiction of how the impairments map to the end user perceived quality, the ITU-T recommended E-Model was used.

The E-model is a computational model for estimating the subjective quality of a VoIP call. It is standardized by the ITU-T (International Telecommunications Union Technical standards) as G.107 [11]. The E-Model combines loss and delay impairments based on the concept that perceived quality impairments are additive. The primary use of the E-model is in the design of codecs and transmission networks. The output of the E-model algorithm is a scalar rating of call quality called the R value.

The R value is calculated from:

$$R = Ro - Is - Id - Ie + A$$

Ro : Basic signal-to-noise ratio
Is : Impairments simultaneous to voice encoding
Id : Impairments due to network transmission
Ie : Effects of equipment (e.g. low bit rate & loss)
A : Advantage factor

Id and Ie are the only variable parameters affected by network transmission. Packet loss is accounted for within the E-model by the equipment impairment factor (Ie) which is loss and codec dependent, while Id is dependent on delay and jitter. Hence, using the metrics obtained by the VoIP application, an E-Model R value can be obtained.

The E-model output *R* can be converted into the more commonly known metric MOS (Mean Opinion Score) which is more commonly used to measure how a user rates call quality. Table 1 shows how the E-model *R* rating maps to the mean opinion score.

Table 1 - VoIP Call Ratings

R value (lower limit)	MOS (lower limit)	User Perception
90	4.34	Very Satisfied
80	4.03	Satisfied
70	3.6	Some users dissatisfied
60	3.1	Many users dissatisfied
50	2.58	Nearly all users dissatisfied

The outputs values from the MH VoIP application are used as the input parameters for the E-Model algorithm, the output of which is used to calculate MOS. The results from a typical experiment are shown in Figure 6, which shows a moving average of the RTT, jitter and MOS. As can be seen when a handover is triggered, SIGMA performs a seamless handover with no increase in delay or jitter. Also, it is worth noting that there is no packet loss during handover. The MOS obtained using the delay, jitter and loss metrics shows that the maximum attainable MOS score of 4.4 for G.711 is maintained during handover. This shows that SIGMA is capable of performing seamless VoIP handovers with no reduction in voice quality.

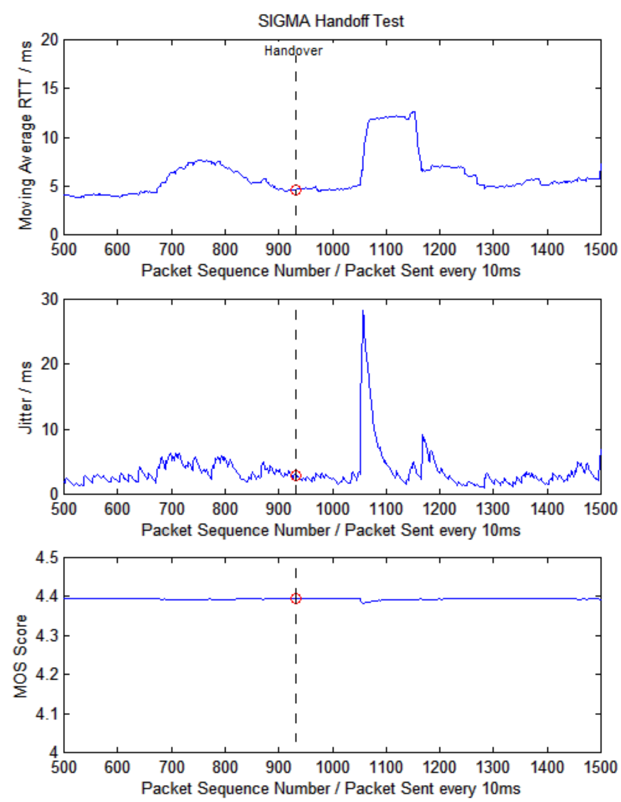

Figure 6 - Delay, Jitter & MOS during Handover

The previous results (Figure 6) show that there is no degradation in voice quality during handoff when the SIGMA handover scheme is used. We now look at the handover latency of SIGMA. The handoff latency is defined as the time between the MH transmitting the ASCONF message and the time that all traffic is being transmitted over the new link. In SIGMA, the handoff latency is directly related to the end-to-end delay between the MH and CN.

Figure 7 shows the handoff latency for a SIGMA handoff obtained using the Wireshark network analyser tool [10]. The signalling diagram shows VoIP data flow and the handover signalling. Each link is identified by an IP address. 10.1.8.5 is the initial primary path through AP1, 10.1.6.5 is the secondary path through AP2, and 129.15.78.118 is the IP address of the CN.

As can be seen when a handover is triggered, the MH sends an ASCONF message to the CN via AP1. This informs the CN to change the current primary address. On reception of the message, the CN modifies the primary and transmits an acknowledgment message. It can be seen in the Figure 7 that for this experiment the handoff latency was 17ms. This value was measured using Wireshark traces and is the time between the ASCONF chunk being transmitted by the MH until all data flows over the new link. The value of 17ms obtained is well below the threshold of 150ms, above which voice quality begins to degrade.

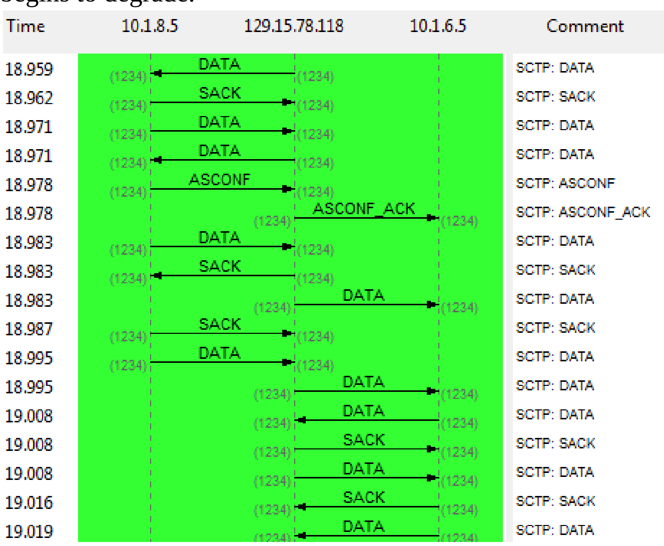

Figure 7 - Flow Graph during SIGMA handover

VI. CONCLUSION & FUTURE WORK

In this paper, the ability of the SIGMA mobility solution to support seamless VoIP handoffs was investigated. A Client/Server VoIP emulating application was developed on a Linux testbed using SIGMA handoff. The Linux kernel distribution of SCTP with the partial reliability extension was used as the transport layer protocol. The application emulated full duplex G.711 VoIP calls over RTP and collected packet level metrics such as delay, jitter and loss.

The testbed consisted of two overlapping 802.11 WLAN access points over which a handover was performed. Results were presented showing the delay, jitter, loss and MOS of a VoIP call during a handover. The results show that SIGMA had a handover latency of 17ms and suffered no packet loss. We have shown that PR-SCTP can be used as a transport layer protocol for VoIP and that SIGMA is capable of providing seamless handovers for VoIP with no degradation in call quality.

ACKNOWLEDGMENTS

The support of the Irish Research Council for Science, Engineering and Technology (IRCSET) is gratefully acknowledged. The work of M. Atiquzzaman was supported by NASA grant NNX06AE44G.

REFERENCES

[1] C E Perkins & A Myles “Mobile IP" IEEE Communications Magazine, May 1997, Volume 35.
[2] Handley, M, "Why the Internet only just works" BT Technology Journal , July 2006, Volume 24 No 3.
[3] Fathi, H. Chakraborty, S. Prasad, R. "Mobility management for VoIP: evaluation of mobile IP-based protocols" IEEE International Conference on Communications, ICC 2005.
[4] Nakajima, N. Dutta, A. Das, S. Schulzrinne, H. "Handoff delay analysis and measurement for SIP based mobility in IPv6" IEEE International Conference on Communications, ICC 2003.
[5] Castelluccia, C. "HMIPv6: A hierarchical mobile IPv6 proposal" ACM SIGMOBILE Mobile Computing and Communications Review, 2000, Vol. Volume 4.
[6] G. Dommety et. al "Fast Handovers for Mobile IPv6" IETF draft, draft-ietf-mobileip-fast-mipv6-03.txt, July 2001.
[7] Torrent-Moreno, M. Perez-Costa, X. Sallent-Ribes, S. "A performance study of fast handovers for mobile IPv6" Local Computer Networks, 2003. LCN '03.
[8] R. Stewart, M. Ramalho, Q. Xie, M. Tuexen, I. Rytina, M. Belinchon,. "Stream Control Transmission Protocol (SCTP) Dynamic Address Reconfiguration" draft-ietf-tsvwg-addip-sctp-07.txt, February 2003.
[9] R. Stewart, M. Ramalho, Q. Xie,M. Tuexen, and P. Conrad "SCTP Partial Reliability Extension" draft-stewart-tsvwg-prsctp-00.txt. 2002.
[10] Wireshark Network Analyser, http://www.wireshark.org/
[11] ITU-T "The E-Model, A Computational Model In Use In Transmission Planning”, G.107, March 2003
[12] Kwon, T.T. Gerla, M. Das, S. "Mobility management for VoIP service: Mobile IP vs. SIP" IEEE Wireless Communications, Volume9, Issue 5, Oct 2002.