Measurement Based Analysis of the Handover in a WLAN MIPv6 Scenario

*Albert Cabellos-Aparicio[1], René Serral-Gracià[1],
Loránd Jakab[2], and Jordi Domingo-Pascual[1]

[1] Universitat Politècnica de Catalunya (UPC),
Departament d'Arquitectura de Computadors, Spain,
{acabello,rserral,jordid}@ac.upc.edu
[2] Universitatea Tehnică din Cluj-Napoca,
Facultatea de Electronică şi Telecomunicaţii, Romania,
moriarty@bel.utcluj.ro

Abstract. This paper studies the problems related to mobile connectivity on a wireless environment with Mobile IPv6, specially the handover, which is the most critical part. The main goal of this paper is to develop a structured methodology for analyzing 802.11/IPv6/MIPv6 handovers and their impact on application's level. This is accomplished by capturing traffic on a testbed and analyzing it with two applications developed for this purpose. The analysis covers passive and active measurements. This methodology is applicable for measuring improvements on handover (such as Fast Handovers for Mobile IPv6, Hierarchical Mobile IPv6 or 802.11 handover).

1 Introduction

A great interest exists among users, in being on-line, permanently and without wires. On the last years, wireless technologies have improved and made cheaper. With WLAN (IEEE 802.11) [1] as one of the most used, it is possible to provide connectivity and bandwidth in a cheap and easy way.

This technology is able to provide ”nomadism” to the Internet, in other words, an user can be connected to the Internet using WLAN, but he can't move, change his point of attachment and maintain his network connections. For that reason, IETF has designed Mobile IP, which, jointly with WLAN, provides this capability to the Internet (this is commonly known as mobility). In this paper, we focus on active and passive measurements using Mobile IPv6 with 802.11b.

The most critical part of these technologies is the handover. It is important to note that during this phase, the mobile node (MN) is not able to send or receive data, and some packets may be lost or delayed (due to intermediate buffers). This lack of connectivity can affect some applications, especially streaming or real-time, which do not have retransmission mechanisms.

* This work was partially funded by the MCyT (Spanish Ministry of Science and Technology) under contract FEDER-TIC2002-04531-C04-02 and the CIRIT (Catalan Research Council) under contract 2001-SGR00226.

This paper focuses on measurements (active and passive) of the WLAN/ IPv6/ MIPv6 handover. Our goal is to study the handover in a real testbed using two different approaches. First, using passive measurements, analyzing the handover latency (the time where the mobile node is not able to send and receive data). Our aim is to compare layer 2, layer 3 and MIPv6 handover, and to find bottlenecks. Secondly, with active measurement; our goal is to study the effects of the handover on traffic sent or received by applications, studying differences depending on flow directions, packet losses, one-way delays and IPDV (IP Delay Variation).

Several papers focus on the same topic, [5] uses a mathematical model to study the handover latency but it does not take into account the wireless handover, [4] studies the Mobile IPv6 (and others) handover with a simulator, [2] makes an empirical analysis of the 802.11 handover, and, finally, [3] studies the WLAN/Mobile IPv6 handover in a real testbed proposing a new algorithm to improve the handover latency. Our paper goes further, analyzing bottlenecks, comparing the layer 2 and layer 3 handover and studying the effects suffered by the applications.

The reminder of this paper is organized as follows: section 2 and 3 are a summary of IEEE 802.11 and Mobile IPv6. Our measurement scenario is presented in section 4. In section 5 we propose an active and passive measurement methodology for handovers, in section 6 we present the results obtained in our handover analysis and finally, section 7 is devoted to the conclusions of the paper.

2 IEEE 802.11

This protocol is based on a cellular architecture, where the system is divided into cells. Each cell (Base Service Set or BSS) is managed by a Base Station (commonly known as Access Point or AP). WLAN can be formed by a single cell (or even by none, in "ad-hoc" mode) but, usualy is formed by a set of cells, where AP's can communicate trough a backbone (Distribution System or DS). All this entities, including different cells, are viewed as a single 802.11 LAN from upper layers (in the OSI stack).

AP's announce their presence using "Beacon Frames" that are sent periodically. When a STA desires to associate to an AP, it has to search for one (scan). Scan can be performed using two different methods, either passive scanning, where STA "listens" for a "Beacon Frame" (which includes all related information to get associated), or active, where STA sends "Probe Requests" frames, expecting to receive "Probe Response" sent by AP's.

Once a STA has found an AP, and decided to join it, it will go through the "Authentication Process", which is the interchange of security information between the AP and the STA. When the STA is authenticated, it will start the "Association Process", AP and STA will exchange information about capabilities and allow the DS to know about the current position of the station. Only after the association process is completed, the STA is able to transmit and receive data frames.

If the signal received by the STA degrades (possibly because has moved away from the AP) the handover procedure starts. First, STA must find a new AP; this is accomplished using ”scan” (described previously). When a new AP is found and the STA decides to join it, it must ”Reauthenticate” and ”Reassociate”.

3 Mobile IP

Mobile IP was designed by IETF in two versions Mobile IPv4 [8] and Mobile IPv6 (MIPv6) [9]. The main goal of the protocol is to allow mobile nodes to change its point of attachment to the Internet while maintaining its network connections. This is accomplished by keeping a fixed IP address on the mobile node (Home Address or HAd). This address is unique, and, when the mobile node is connected to a foreign network (not its usual network) it uses a temporal address (Care-of Address or CoA) to communicate, however, it is still reachable through it's HAd (using tunnels or with special options in the IPv6 header). In this paper, we focus in MIPv6, although the tools developed can be easily migrated to MIPv4, FastHandovers for Mobile IPv6 [10], or other handover improvements.

MIPv6 has three functional entities, the Mobile Node (MN), a mobile device with a wireless interface, the Home Agent (HA), a router of the home network that manages localization of the MN, and, finally the Correspondent Node (CN), a fixed or mobile node that communicates with the MN.

The protocol has four phases:

1. *Agent Discovery:* The MN has to discover if it is connected to the home network or to a foreign one. For this purpose uses ”Router Advertisements” [11], those messages are sent periodically by all IPv6 routers and include information for client autoconfiguration. Using this information, the MN obtains a CoA.
2. *Registration:* The MN must register its CoA to the HA and to CN's. This way, they know ”who” is the MN (HAd) and ”where” it is (CoA). Some messages related to this phase are ”Binding Update” (BU) and ”Binding Acknowledgment” (BA).
3. *Routing and Tunnelling:* MN establishes a tunnel with the HA (if it is necessary), and it is able to receive and send data packets (using the tunnel, or directly).
4. *Handover:* MN changes its point of attachment. It must discover in which network it is connected (phase 1) and register its new CoA (phase 2). During this phase, some data packets (sent or received by the MN) can be lost or delayed due to incorrect MN location.

4 Measurement Scenario

This section describes the practical part surrounding the setup of a measurement scenario for networking tests. Also describes the different hardware and software used for all the tests shown in this paper.

4.1 Network topology

Testbed's main reason is to compute Mobile Node handover latencies, the testbed in detail can be seen in Figure 1, there are all the different parts of the scenario. To avoid external interferences, this testbed is isolated from outside networks, all the input and output traffic on the testbed's network interfaces is controlled. Having this isolation, but without the lack of external access, the scenario has two parallel networks, the control network and the actual testing network, as highlighted on the figure. This is important because with uncontrolled sources of traffic all the delays will be miscalculated.

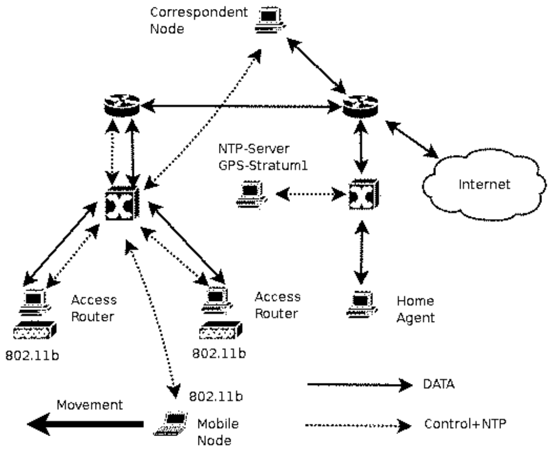

Fig. 1. Measurement scenario simplified structure

This testbed gives the tests all the privacy needed, this way, once the tests are prepared, no foreign agents are able to interfere with them. At the same time, the path followed by the packets is long enough to consider the possible clock skew too small to have any negative impact on the results.

Regarding synchronization, the testbed is configured to use four NTP (Network Time Protocol) sources [12], two of them belonging to a private network, Stratum 1 servers connected to a GPS source each. The other two sources are on the outside network and are as far as 3 hops away from the testbed. All the NTP traffic is routed through a parallel network (with the local NTP servers) where there isn't any other traffic. It is possible to access those remote NTP servers through the control network which can use external time sources. The NTP statistics shows that, with this setup, we obtain 1ms of measurement accuracy.

In order to confirm our synchronization accuracy, we made a simple test; we sent several ARP broadcast packets in our measurement network, those packets were captured on all the machines involved in our tests, and the timestamps were compared. The maximum difference among those timestamps agreed with the threshold stated by NTP.

4.2 Hardware and software equipment

Depending on the testbed description, all the machines involved on the tests are using the GNU/Linux Debian Sid distribution. Depending on the role of each computer, the hardware and the kernel varies accordingly:

- *Access Points/Routers (AP):* this testbed has two access points, each one with two wireless cards, one for communicating with the MN and the other one to monitor (capture frames). Those cards have the Atheros Chipset (802.11g) in 802.11b compatibility mode. The configured kernel is the 2.4.26.
- *Mobile Node (MN):* this mobile node uses a Cisco Aironet 350 card (802.11b) for wireless connectivity, here the kernel is 2.4.26 with MIPv6 1.1 [14] patch for Mobile IPv6 support.
- *Home Agent (HA)/Correspondent Node (CN):* the last two important hosts on the scenario have similar configuration with the 2.4.26 kernel patched the same way as the Mobile Node for Mobile IPv6 capabilities.

5 Methodology

This section is devoted to the description of the methodology developed for this paper. As our goal is to analyze the handover, we chose several tools which permit to measure the desired network parameters. Those tools are:

1. *MGen/DRec, NetMeter [15]:* for the active measurement part.
2. *Ethereal [13] and PHM Tool (Passive Handover Measurement) [6]:* for passive measurements.

Both applications depicted here: NetMeter's handover analysis module and PHM Tool are developed under the same code base. Their main goal is to analyze the Ethereal files and obtain for PHM Tool the handover latencies and for the NetMeter's part the packet losses and delays at application level.

The same capture is used for both solutions, the monitoring infrastructure is set up on the Access Points, given that is the only way of detecting all the handover latencies. Both captures (each on one access point) are merged (as they really represent the same traffic flow) and the data is analyzed.

The following subsection enumerates the set of tests prepared for this paper, following with the description of the *passive* analysis and later the paper focuses on the study of the *active* part.

5.1 Tests

For a good analysis of the handover, is necessary to build up a good set of tests. In this paper we ran a set of 16 tests, each 5 minutes long, from where extracted a set of 63 valid handovers.

Half the tests had the generated traffic from the Correspondent Node to the Mobile Node, while the other half was on the opposite direction.

Moreover, each direction of the tests where split as follows:

- *64Kbps Traffic*: this flow simulates with UDP the properties of VoIP traffic under IPv6, there are sent 34 packets per second with 252 bytes of payload as stated on [7].
- *1Mbps Traffic*: due to the low rate needed for VoIP the other tests are done on a higher packet rate, so the impact of a different bandwidth can be studied. This time there were 94 packets per second with a payload size of 1300 bytes per packet.

The VoIP simulation was chosen because all the traffic constraints of such technology are well known and will be easy to determine the user impact of the handover on such traffic regarding delays and packet losses.

5.2 Passive Handover Measurements (PHM)

Our main goal is to measure the handover latency or, in other words, the amount of the time where a MN is not able to send or receive data. This duration has several components, the amount of time spent by layer 2 (802.11b in our case) in scanning for a new AP, authenticate and re-associate to it, time used by IPv6 on connecting to the new network and, finally, the amount of time spent by MIPv6 in registering it's new CoA to HA and CN's.

The developed application "PHM Tool" monitors the signaling messages in both AP of our testbed. We capture all packets sent or received by their wireless interface. Handovers are "forced" attenuating the signal sent by the AP. The MN realizes this (it detects that the signal quality is poor) and tries to search for a new AP. In our testbed we do not have external interferences, and thus, the MN changes to the other AP.

When a set of handovers have been carried out, the captured packets are processed off line using "PHM", which analyzes the signaling messages providing results.

5.3 Active Handover Measurements

Usually, pure Active Measurements, have an end-to-end approach. The basis of such tests is to generate a synthetic flow travelling through the network under test. This paper proposes a new method for enhancing the Active Measurement framework. Our approach is based on a mixed use of Passive and Active Measurement systems. The whole point is to generate the Active flow and measure the typical end-to-end parameters. This flow is captured at its destination, but also at the Access Point (using typical capture software such as Ethereal [13]).

Once the tests are finished, the captured data is converted to the standard XML language for network packets (PDML [16]), this file is processed by our analysis tool, which will convert the data to a standard MGen file. This approach permits to calculate partial data delays, that's because the stored timestamps passed to the MGen file are taken from the monitoring machine (which is the actual Access Point on the testbed). Our solution can be used on a wide variety of scenarios on general network measurement systems.

Focussing now on the paper's tests, our method is to isolate the parameters computation on the wireless data flow from the wired one. This way, is possible to isolate all the handover incidences without taking into account the other parts of the tests. Besides, another possible use of the testbed on such conditions is to model the impact on the user's perception of the packet losses and bigger delays caused by the handover, this time with the end to end results.

6 Results

This section describes the results obtained from the tests discussed on the previous section. First the discussion focuses on the passive set of handovers for analysing its duration and all the parts throughout the process, later the analysis of the active results and the user level performance are shown.

6.1 Passive Handover Measurements

The whole system was tested doing a set of handovers, capturing all the signaling messages and processing them off line using PHM.

Table 1. Numerical results obtained using PHM (ms)

	Mean	**Std. Dev.**
Scan	257.224	108.007
Authentication	2.733	1.183
Association	1.268	0.311
IPv6	1836.413	430.196
Registration (HA)	3.914	1.017
Registration (CN)	9.262	4.881
Total time	2107.82	450.619

The table 1 (results in milliseconds) show the results obtained with our application and are a detailed version of the handover latency, and reveal time between two consecutive signaling messages. Wireless handover is detailed and we can see that the scan phase is the longest one; the MN uses in average 257ms to find a new AP. The whole 802.11 handover (Scan, Authentication and Association) represents 12% of the total handover latency.

The second phase is the time spent by IPv6 to realize that it is attached to a new network, and obtain a new CoA. IPv6 uses more than one second because it has to perform DAD (Duplicate Address Detection) and to realize that its old default router (the previous AR) it is unreachable [11]. For this last action, the

MN has a timeout, if this was set to a very low value (less than one second) the MN, while communicating, would be sending ”Router Solicitation” messages constantly trying to know if its default router is present or not. In our testbed, this timeout was set to its minimum value. This part is 87% of the total time.

The Mobile IPv6 handover is also detailed, the first part (Registration HA) is the time spent by the MN to indicate to the HA its new location, the second part (Registration CN) is the time used by the MN to announce its new point of attachment to the CN. It takes more time just because authentication between MN and CN includes more messages. This part is 1% of the total handover latency. This time is related to the round-trip time to the HA and to the CN.

We can conclude that in an 802.11/IPv6/MIPv6 handover, most part of the time is due to IPv6 (87%), and specifically, due to ”Neighbor Unreachability Detection”, the algorithm used to detect if the default router is present or not.

6.2 Active Handover Measurements

Once the analysis of the low level handover is finished, the next step on our study is to analyze the traffic’s impact on user level, here the most important parameters are: packet losses and delays. As will be discussed, another important result depicted here is the differences on the results regarding the traffic direction, from the CN to the MN or the other way around.

A - Packet Losses

To compute packet losses is a straight forward problem having the first and the last packet involved on the handover, which are provided by the capture on the access point.

Either Figure 2 and 3 have the packet’s sequence number on the x-axis and the delay (in milliseconds) on the y-axis.

Figures 2 and 3 show two different handovers, the first one represents the flow going from the Correspondent Node to the Mobile Node, and the other one the opposite flow direction. Both handovers have similar duration, but, different packet losses (due to different rates) as shown in the table 2.

Table 2. Packet Losses

	Mean		**Std. Dev.**	
	64Kbps	**1Mbps**	**64Kbps**	**1Mbps**
MN -> CN	65.80	162	9.78	16.97
CN -> MN	61.71	207.21	17.54	65.90

Table 2 shows the summary of mean and standard deviation of packet losses per handover on the whole set of tests done. As can be seen the higher is the

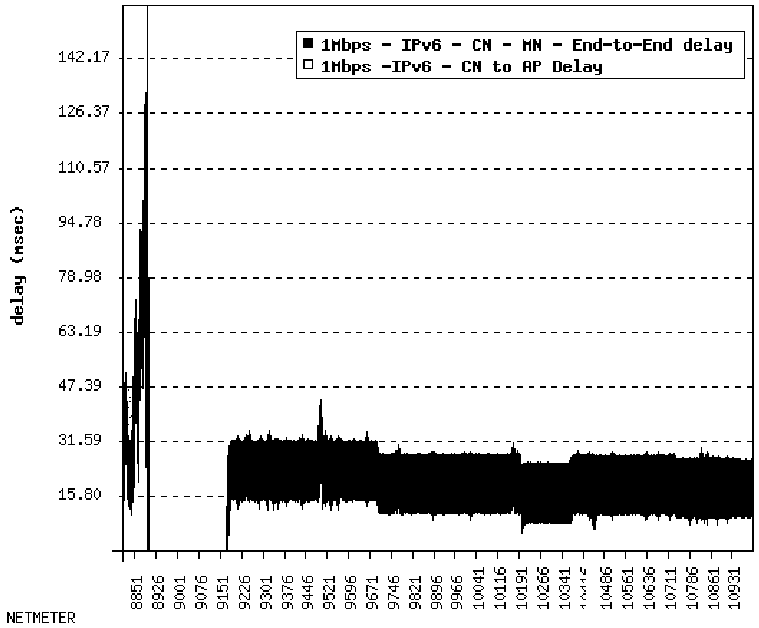

Fig. 2. Handover from CN to MN with 1Mbps traffic

packet rate higher is the packet's loss. The mean duration for the handovers treated on this part (at application's level) is ***1,98 sec***, which is slightly different from the value given on the passive analysis, that's because of a different set of analyzed handovers.

B - Flow directions

Regarding the effects suffered by a traffic flow when it is send from the MN to the CN or vice versa table 2 doesn't show any difference. However, figures 2 and 3 show that the handover has slight differences.

When the MN is sending packets to the CN and a handover starts, it stops immediately while searching for a new Access Point, those packets are buffered at Layer 2 by the wireless drivers. After the 802.11 handover is finished, the MN starts to send those buffered packets as fast as it can, but they are lost because they are sent to the old Access Router. Only after the MIPv6 handover is finished, the packets flow correctly to the CN. Figure 3 shows this behavior, packets marked as "End-to-End delay" (those are the packets that arrive correctly to the CN) reveal the handover gap while packets marked as "MN to AP delay" (not all those packet arrive correctly to the CN) show that during the handover some of them are buffered and sent after the 802.11 handover is finished with a higher delay, specifically the maximum delay is the duration of the wireless handover.

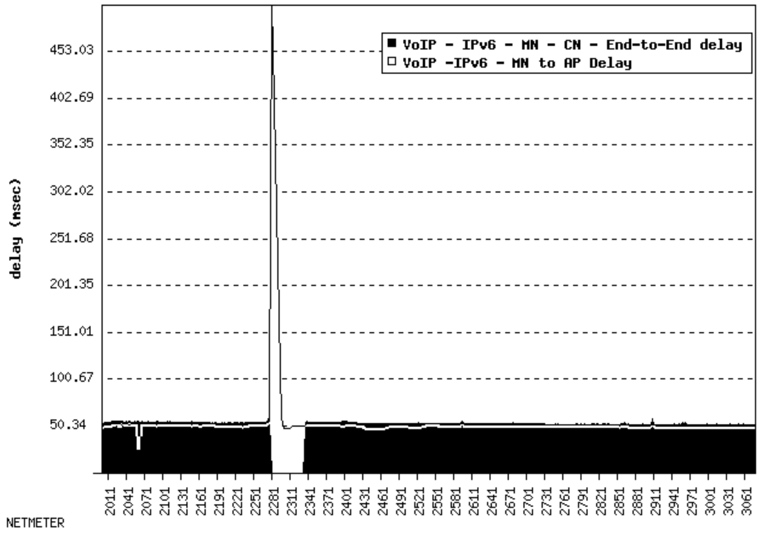

Fig. 3. Handover from MN to CN with VoIP traffic

The MN's buffer may seem useless, but, in fact, is very effective in case of an 802.11 handover. In this case, the MN is not changing its default router, it doesn't need to send a Binding Update indicating a new location, it is just changing it's AP. The buffer will store packets that otherwise would be lost, those packets will be sent correctly (but with a higher delay) when the MN regains Layer 2 connectivity.

In the case that the CN is the source (figure 2) of the traffic flow the handover behaves differently. The CN sends packet constantly (it is connected to an Ethernet), when the handover starts, all those packets are lost because they are sent to the incorrect Access Router. Only after the MIPv6 handover part is finished, the CN realizes of the new location of the MN and sends the packets to the correct address.

C - QoS parameters consideration

Under a QoS environment, as stated above, there are other important parameters which highlight the level of provided QoS. Those parameters are the **One Way Delay** (OWD) and the **Inter Packet Delay Variation** (IPDV). Much discussion is possible on this subject, but only for the sake of simplicity, the study will be limited to the handovers studied on the previous figures, the statistically representative study is left as future work. The goal is to see if the QoS parameters are kept under those handovers. This is accomplished by taking near three seconds worth of packets before the handover and calculate the OWD, the IPDV and the same after it. With this is possible to see if there are grave variation of the above parameters when the Access Point signal's quality decreases just before the handover, or instead, if when associated to the new

Access Point the system's convergence time to the new configuration causes any more problems.

Table 3. OWD and IPDV

	OWD (ms)		**IPDV** (ms)	
	Before	**After**	**Before**	**After**
VoIP	54.08	53.25	0.0125	0.0096
1Mbps	63.58	28.51	7.2566	-0.0011

The overall results for the displayed handovers can be seen on table 3, the results are very clear, when there is low traffic on the wireless link (VoIP), the loss of connectivity before the handover, hardly affects the packet delays, the same holds true for the system recovery once the handover is finished.

Another result, though, is the case when the link is more overloaded (1Mbps), where is easy to see the increment on the delivery delays of the flow, the reason is the loss of link quality on the wireless link, although the system's recovery is pretty fast and reliable. The same results can be seen on Figures 2 and 3.

7 Conclusions

This paper analysis focuses on all levels involved in the handover process, from 802.11 handover until application's level. That's why we designed a testbed and developed two applications to make active and passive handover measurements.

Passive measurements are intended to compute the handover latency in order to find bottlenecks and to compare layer 2, layer 3 and MIPv6. In the other hand, we expect to compute important parameters such as delay, IPDV and packet losses with active measurements in order to analyze the impact at application's level forced by such handovers.

Passive results show that an 802.11/IPv6/MIPv6 handover takes 2.107 seconds in average. The 802.11 part is 12% of the total time; most of this time is spent searching for a new AP. The IPv6 part is the longest one, takes 87% of the total time, the MN has to realize that its previous default router is no longer reachable and switch to the new one. Finally, the MIPv6 part is 1% of the total time.

Summarizing all the obtained results for the active measurement, the handover as is doesn't forbids the QoS on low bandwidths in terms of one way delay. The problem, though, is uncovered by the packet losses (which is proportional to the handover latency), where its value, depending on the packet's rate is about 63 losses per handover (VoIP), which is unbearable for a proper quality voice transmission. The only solution for this matter is to improve the handover times, that is, to improve Mobile IPv6, or change it to better protocols such as Fast Handovers.

Mixing both worlds (passive and active measurements paradigm) opens up a new set of possibilities for analyzing all the different aspects of the handover. We plan to extend this handover analysis, using the same methodology, to other protocols such as Mobile IPv4, Fast Handovers for Mobile IPv6, Hierarchical Mobile IPv6 or some IEEE 802.11 handover improvements. We also want to extend the methodology in order to know, exactly, how many packets, and which packets have been lost or delayed in a given handover phase, this will be useful for protocols such as Fast Handovers that uses intensively buffering.

References

1. IEEE: 802.11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specification Arch. Rat. Mech. Anal. (1997)
2. A. Mishra, M. Shin and W. Arbaugh: An Empirical Analysis of the IEEE 802.11 MAC Layer Handoff Process **Volume 33** *ACM SIGCOMM Computer Communications Review* (2003)
3. N. Montavont and T. Noel: Handover Management for Mobile Nodes in IPv6 Networks *IEEE Communications Magazine* (2002)
4. Xavier Pérez Costa and Hannes Hartenstein: A simulation study on the performance of mobile IPv6 in a WLAN-based cellular network *40 Issue 1 Computer Networks: The International Journal of Computer and Telecommunications Networking* (2002)
5. Marco Liebesch, Xavier Pérez Costa and Ralph Schmitz: A MIPv6, FMIPv6 and HMIPv6 Handover latency study: Analytical Approach *IST Mobile and Wireless Telecommunications Summit* (2002)
6. Loránd Jakab, Albert Cabellos-Aparicio, René Serral-Gracià, Jordi Domingo-Pascual: Software Tool for Time Duration Measurements of Handovers in IPv6 Wireless Networks *UPC-DAC-2004-25* (2004)
7. John Q. Walker, NetIQ Corporation: A Handbook for Successful VoIP Deployment: Network Testing, QoS, and More (2002)
8. C. Perkins: IP Mobility Support for IPv4 *RFC 3344* (2002)
9. D. Johnson, C. Perkins and J. Arkko: IP Mobility Support for IPv6 *RFC 3775* (2004)
10. Rajeev Koodl: Fast Handovers for Mobile IPv6 *draft-ietf-mipshop-fast-mipv6-03.txt* (2004)
11. T. Narten, E. Nordmark and W. Simpson: Neighbor Discovery for IP version 6 (IPv6) *RFC 2461* (1998)
12. Internet2 Consortium: OWAMP - NTP Configuration *http://e2epi.internet2.edu/owamp/details.html#NTP* (2004)
13. Gerald Combs: Ethereal: The world's most popular network protocol analyzer *http://www.ethereal.com* (2004)
14. Helsinki University of Technology: MIPL Mobile IPv6 for Linux *http://www.mobile-ipv6.org/* (2004)
15. René Serral, Roberto Borgione: NetMeter a NETwork performance METER *http://www.ccaba.upc.es/netmeter* (2002)
16. PDML Specification: *http://analyzer.polito.it/30alpha/docs/dissectors/PDMLSpec.htm* (2002)