Measurement-Based Comparison of IPv4/IPv6 Mobility Protocols on a WLAN Scenario

Albert Cabellos-Aparicio, Hector Julian-Bertomeu, Jose Núñez-Martínez, Loránd Jakab,
René Serral-Gracià, Jordi Domingo-Pascual
Universitat Politècnica de Catalunya (UPC),
Departament d'Arquitectura de Computadors, Spain
{acabello,bertomeu,jnunyez,ljakab,rserral,jordid}@ac.upc.edu

Abstract

This paper focuses on a measurement-based comparison of the handover for different mobility protocols: Mobile IPv4, Mobile IPv6 and Fast Handovers for Mobile IPv6. The paper studies the handover using active and passive measurements computing the handover latency, packet losses and the provided QoS level in a real testbed. Our experimental results show that there are severe QoS fluctuations before the handover for Mobile IPv4 and Mobile IPv6, moreover Mobile IPv6 has unacceptable handover latency for real time traffic. Our implementation of Fast Handovers for Mobile IPv6 behaves as expected, reducing the Mobile IPv6 handover latency, delaying packets instead of losing them and maintaining the level of provided QoS before and after the handover.

Key Words: *Mobility, Measurement, Mobile IPv4, Mobile IPv6, Fast Handovers, QoS, Handover*

1. Introduction

Wireless technologies have evolved in recent years. IEEE 802.11 [1] is one of the most used wireless technologies and it provides up to 54Mbps of bandwidth in an easy and affordable way. In current Internet status, an user can be connected through a wireless link but he cannot move without breaking the IP communications. That's why IETF designed Mobile IP which jointly with IEEE 802.11 provides mobility to the Internet. With "mobility" a user can move and change his point of attachment to the Internet without losing the network connections because he will have a fixed IP address regardless of his location.

The most critical part of this technology (IEEE 802.11 and Mobile IP) is the handover. The handover is the time spent when changing from one point of attachment (i.e. an access router) to another. During this time the mobile node is not able to send or receive data and thus, the connection may be interrupted, packets may be lost or delayed due to intermediate buffers.

This paper focuses on handover measurements. We compare the handover of different mobility protocols in a real testbed: Mobile IPv4, Mobile IPv6 and Fast Handovers for Mobile IPv6. We study the handover using different approaches. First we compute the handover latency, in other words: the duration of the handover. Second we study how many packets are lost during the handover for the different mobility protocols, taking into consideration if the packets are being sent from the mobile node or to the mobile node. Finally we study the QoS level provided before and after the handover. We use public available implementations of Mobile IPv4 and Mobile IPv6 for the GNU/Linux O.S and we have developed a Fast Handovers for Mobile IPv6 [10] implementation (under the GPL license) that, as far as we know is the first public available existing implementation.

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.

Several papers focus on the same topic. [2] uses a mathematical model to study the handover latency without taking into account the IEEE 802.11 handover. [3] studies the Mobile IPv6 handover not in a real testbed but in a simulator. [4] makes an extended IEEE 802.11 handover study using different wireless cards. [5] proposes a new algorithm to improve the IEEE 802.11/Mobile IPv6 handover latency analyzing it in a real testbed. Finally [6] studies the signalling overhead and other parameters of the Fast Handovers protocol through a simulator.

Our paper's main contributions are a comparison of Mobile IPv4, Mobile IPv6 and Fast Handovers for Mobile IPv6. The measurement methodology and some preliminary results of the Mobile IPv6 protocol have been published in [7]. The FMIPv6 implementation and a performance evaluation have been presented in [21].

The remainder of this paper is organized as follows: section 2 presents an overview of the wireless and mobility protocols involved in the study. The measurement scenario and the methodology is presented in section 3, while section 4 presents the results of the handover mobility measurements and an extended comparison among them. Finally, section 5 is devoted to the conclusions of our study.

2. Mobility Protocols Overview

2.1. IEEE 802.11

The Wireless LAN protocol [1] is based on a cellular architecture, where each cell is managed by a Base Station (BS, commonly known as Access Point or AP). Such a cell with the BS and the stations (STA) is called a Basic Service Set (BSS) and can be connected via a backbone (called Distribution System or DS) to other cells, forming an Extended Service Set (ESS). All these elements together are one single layer 2 entity from the upper OSI layers' point of view. APs announce their presence using periodic "Beacon Frames" containing synchronization information. If a STA desires to join a cell, it can use passive scanning, where it waits to receive a "Beacon Frame" or active scanning, when it sends "Probe Request" frames and receives a "Probe Response" frame from all available APs. Scanning is followed by the Authentication Process, and if that is successful, the Association Process. Only after this phase the STA is capable of sending and receiving data frames. STAs are capable of roaming, i.e. moving from one cell to another without loosing connectivity but the standard does not define how it should be performed, it only provides the basic tools for that: active/passive scanning, re-authentication and re-association.

2.2. Mobile IP

Mobile IP was designed in two versions, Mobile IPv4 (MIPv4) [17] and Mobile IPv6 (MIPv6) [18]. The protocol's main goal is to allow MNs to change its point of attachment to the Internet while maintaining its network connections. In other words, the mobile node has a special IP address (Home Address or HAd) that will remain unchanged regardless of the MN's location, moreover, the MN will use temporary IP Addresses (Care-of-Address or CoA) when connected to foreign networks (not its home network), however, it is still reachable through its HAd (using tunnels or with special options in the IP header). A special entity (Home Agent or HA) manages MN's localization by binding the MN's CoA to MN's HAd.

Mobile IP has three functional entities: the Mobile Node (MN) which is any mobile device with a wireless card and the Mobile IP protocol, the Home Agent (HA) which manages MN's localization and finally the Correspondent Node (CN), a fixed or mobile node that exchanges data packets with the MN. Specifically, Mobile IPv4 has another functional entity: Foreign Agents (FA). A Foreign Agent is a router of a foreign network. IPv6 routers send periodically "Router Advertisement" messages including autoconfiguration and network information but IPv4 routers don't. With those messages a MN can detect if its point of attachment has changed. Mobile IPv4 relies on Foreign Agents which send "Agent Advertisement" messages for configuration and movement detection purposes.

The protocol has four phases. Initially in the *Agent Discovery* phase the MN has to discover if it is connected to its home network or to a foreign one. Either thorough "Router Advertisements" (for IPv6) or through "Agent Advertisements" (for IPv4) the MN will discover to which network it is attached and will obtain a new CoA if it is not in the home network. Next, in the *Registration* phase, the MN must register its CoA (where it is located) to the HA in order that it can bind it with the HAd. This registration is done either with "Registration Request/Reply" messages for IPv4 or with "Binding Update/Acknowledgement" for IPv6 messages. Moreover, in IPv6 a MN can send a "Binding Update" to its CNs to allow direct communication, otherwise packets will be forwarded through the HA. After this phase, *Registration and Tunnelling* comes, the MN establishes tunnels (if necessary) with the HA and CNs in order to send or receive data packets. Notice that the CNs will still send packets to the same destination IP address (the HAd). The last phase is the *Handover*, the MN changes its point of attachment and it must discover in which network it is connected once again (*Agent Discovery*) and register its new CoA (*Registration*). During this phase some data packets can be lost or delayed due to incorrect MN's location.

2.3. Fast Handovers

Fast Handovers for Mobile IPv6 (FMIPv6) [22] is a MIPv6 handover enhancement that reduces the handover latency and stores packets delaying them instead of losing them. This is accomplished by allowing the MN to send packets as soon as it detects a new subnet network link (IEEE 802.11 in our case) and delivering packets to the MN as soon as its attachment is detected by the new access router.

FMIPv6 has different operational procedures, for instance, in the "Predictive Handover" the MN discovers nearby APs using the IEEE 802.11 "scan" and then requests all the important information related to the corresponding new access router. When attachment to an AP takes place, the MN knows the corresponding new router's coordinates including its prefix, IP address and MAC address. Through special "Fast Binding Update" and "Fast Binding Acknowledgment" messages the MN is able to formulate a prospective new CoA (without changing its point of attachment). This CoA must be accepted by the new access router prior to the MN movement. Once the MN has changed its point of attachment and it is connected to the new access router link, it can use its new CoA without having to discover the subnet prefix, it also knows the new access router MAC and IPv6 address, and hence this latency is eliminated. As soon as it is attached, the MN sends a "Fast Neighbor Advertisement" announcing its presence. Moreover, the previous access router will tunnel and forward packets to the new care of address until the MN sends a "Binding Update" registering its new CoA to the HA and the CNs, hence, no packet is lost. The other FMIPv6 operational procedure is the "Reactive Handover" which is very similar to the previous one, however this is not supported by our implementation at the moment.

3. Measurement Methodology

This section presents our measurement scenario and the methodology used to perform the comparison. We will use the methodology presented in [7]. The methodology's objective is to measure the handover for any mobility protocol, we have adapted it to measure the MIPv4 and FMIPv6 handover.

3.1. Measurement Scenario

The testbed's main goal is to study the handover using active and passive measurements. The testbed can be seen in detail in Figure 1.

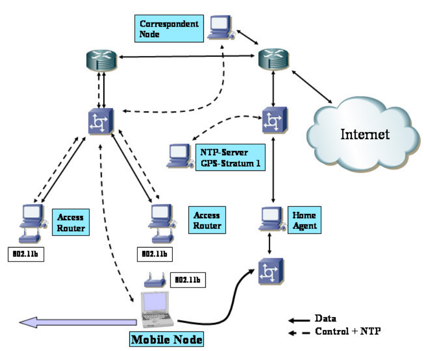


Figure 1. *Simplified measurement scenario*

The testbed has two parallel networks, one for measurement purposes and the other for management traffic and for synchronization. Regarding synchronization the testbed is configured to use four NTP (Network Time Protocol) sources [23], 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. The NTP statistics show that, with this setup, we obtain less than 1ms of measurement accuracy. In order to mitigate the harmful effects of the jitter on the NTP algorithm running in our tetsbed, we use the Pulse-per-Second (PPS) Clock Discipline driver with a PPSAPI interface, which is a proposed IETF Standard [19].

All our GPS sources, like other timekeeping gears such as radio clocks, have a pulse-per-second signal than can be used to force the system clock to a high degree of precision. With PPS we reduce the accumulated jitter and re-time a secondary server when synchronized to a primary server over a congested network. Our machines are typical workstations with low processing and the incidental clock skew and drift can be reduced to a few microseconds.

We use GNU/Linux Debian Sid distribution for our testbed with 2.4.26 Kernel version. We use the MIPL 1.1 Mobile IPv6 [8] and the Dynamics HUT Mobile IPv4 [9] implementation. We have developed a Fast Handovers implementation [10] written for the MIPL 1.1 protocol that complies with the draft-ietf-mipshop-fast-mipv6-03.txt (though not all the features are implemented yet) and that supports any wireless card (with Linux Support) through the "Wireless Toolkit for Linux" [11]. The MN uses a CISCO Aironet 350 card and the access routers have Atheros chip based cards, those access routers are equipped with two cards, one for measurements and the other to monitor and capture layer 2 frames.

3.2. Handover Latency

To compute the handover latency of the different mobility protocols we use passive measurements. All the mobility protocols have different handover parts as explained above. First goes the wireless layer, then the layer 3 (either IPv4 or IPv6) and finally the mobility protocol. All those protocols send their corresponding signalling messages to perform the handover. We capture all these signalling messages sent or received by the MN using a monitoring wireless card. Using the developed application "PHM tool" [10] we compute the different parts of the handover latency of the different protocols offline. "PHM tool" compares the timestamps of the signalling messages providing numerical results of the handover latency. See [7] for further details.

During a handover, first the IEEE 802.11 card detects that the signal quality received by the current Access Point is becoming poor and "scans" for nearby access points sending "Probe Request" messages. This IEEE 802.11 signalling message denotes the handover start. After the wireless card has found a new access point it must authenticate and associate to it, the last message sent by the AP to the MN is the "(Re)Association Reply" that points the end of this part of the handover latency.

With Mobile IPv6, the MN must obtain a new CoA. IPv6 routers send periodically "Router Advertisement" messages which include autoconfiguration information. First the IPv6 layer must check, using the *Neighbor Unreachability Detection* algorithm (NUD) [12] that its previous access router is no longer reachable, then it will listen for "Router Advertisement" messages and it will configure a new CoA. Finally, using the *Duplicate Address Detection* algorithm (DAD) [13] it will ensure that its new CoA is unique on that link and then it will be ready to send and receive IP packets.

Computing the Mobile IPv6 handover latency is a straightforward problem. Mobile IPv6 sends a "Binding Update" to HAs and CNs indicating its new CoA (its new location) and receives a "Binding Acknowledgement" as response. "PHM Tool" computes the Mobile IPv6 handover latency starting at the "Binding Update" message to the HA and ending with the "Binding Acknowledgment" received by the CN.

The Mobile IPv4 implementation has been tuned to speed up the handover as much as possible. The implementation is constantly polling the wireless card and as soon as it detects that the AP has changed it sends a "Router Solicitation" triggering an "Agent Advertisement". This message has all the related information to configure a new CoA. The MIPv4 implementation is also configured to accept this new "Agent Advertisement" without waiting until the last one expires (from the old access router). Next, the MN registers its new CoA sending a "Registration Request" to its HA. The "PHM tool" computes the MIPv4 handover latency from the "Agent Advertisement" to the "Registration Reply".

FMIPv6 enhances the Mobile IPv6 handover reducing the IPv6 handover latency. As explained above, when FMIPv6 is used, the MN prepares the handover to the new access router when it is still connected to the old one. The whole IEEE 802.11/IPv6/FMIPv6 handover latency is reduced to the IEEE 802.11 handover latency. However we must check that our implementation works as expected and "PHM tool" computes the handover latency from the end of the IEEE 802.11 handover until the "Fast Neighbour Advertisement" message.

3.3. Packet Losses and QoS Parameters

We use active measurements to compute the packet losses and other QoS parameters. The basis of active measurements is to generate a synthetic flow travelling through the network under test. We generate an active flow and measure the typical end-to-end parameters; this flow is captured at its destination. NetMeter (see [14] for further details) is the application used for such tests.

To measure the packet losses of the different mobility protocols we send the active flow (either from the CN or from the MN) and force a handover. The number of packets received is computed and the difference between the packets sent and received is the number of losses.

The other important QoS parameters to study the handover are the One-Way-Delay (OWD) and Inter Packet Delay Variation (IPDV) before and after the handover. We use again active measurements to compute them. Having the active flow travelling from the CN to the MN or vice versa we force a handover and we compute the OWD and IPDV for the packets before and after the handover. See [7] for further details.

3.4. Tests

For a good analysis of the handover is necessary to build a good set of tests. In this paper we ran the following set of tests for each mobility protocol under test: Half of the tests had the generated traffic from the CN to the MN while the other half was on the opposite direction. Moreover each direction of the tests was split as follows:

- *VoIP Traffic*: This flow simulates with UDP the properties of VoIP traffic. There are sent 34 packets per second with 252 bytes of payload as stated in [15].

- *Data Traffic*: In order to compare a different bandwidth the other tests are done on a higher packet rate. This flow has 84 packets per second with a payload size of 762 bytes per packet.

We ran a set of 16 tests, each 5 minutes long from where extracted a set of 63 valid handovers for Mobile IPv6 and for Mobile IPv4 we extracted 60 valid handovers. Finally for Fast Handovers we ran a set of 10 tests each 5 minutes long extracting 40 valid handovers.

The FMIPv6 tests are done only from the CN to MN. When the packets flow in this direction, the access routers must tunnel and buffer packets showing an interesting behaviour. However when the traffic source is the MN, there is no need to tunnel packets, just to buffer them on the MN (the FMIPv6 handover latency remains constant for both directions), that's why we focus on the CN to MN direction.

For Mobile IPv4 and Mobile IPv6 the handovers are "forced" attenuating the signal sent by the AP. The MN realizes this (it detects that the signal quality is becoming 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. This procedure tries to simulate a common user movement.

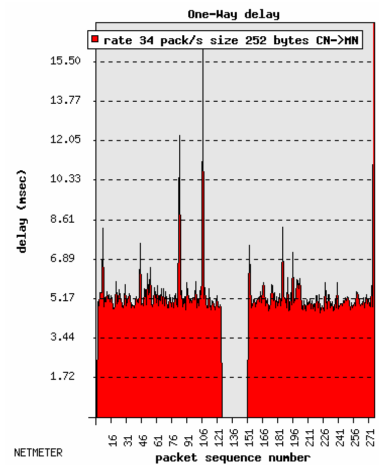
Our Fast Handovers implementation behaves as stated in [16]. When the MN receives the "Fast Binding Acknowledgment" message it is ready to move to the new access router. At that point we force the wireless card to change from the old AP to the new one. As soon as our implementation detects the new link (using [11]) we send the "Fast Neighbor Advertisement" to announce the MN's presence. This method provides a faster handover because the MN does not need to wait until the wireless signal quality becomes poor, in fact, it knows to which AP will move when it is still on the previous link.

4. Results

This section describes the results obtained from the tests discussed on the previous section. The main goal of the results is to compare the protocols and to show trends, they do not must be taken as absolute numbers because they depend on the protocol implementation and the hardware, although the results are statistically representative.

4.1. Overview

This section presents an overview of the obtained results, Figures 2, 3 and 4 show three instantaneous One-Way-Delay (obtained with NetMeter) for the three protocols under test. All the figures have the packet sequence number on the x-axis and the OWD (in milliseconds) on the y-axis. Note that the figures are not in the same scale.



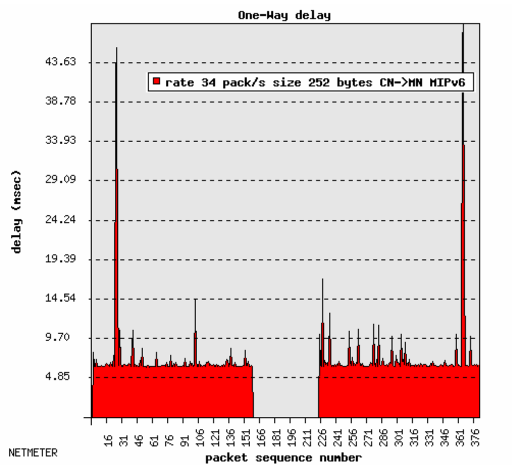


Figure 2. *MIPv4 handover (instantaneous OWD* **Figure 3**. *MIPv6 handover (instantaneous OWD)*

For MIPv4/MIPv6 we can clearly see the gap produced by the handover (where packets are lost). Packet sequence numbers show that the MIPv4 handover latency is lower (and thus the packet losses) than the MIPv6's handover latency.

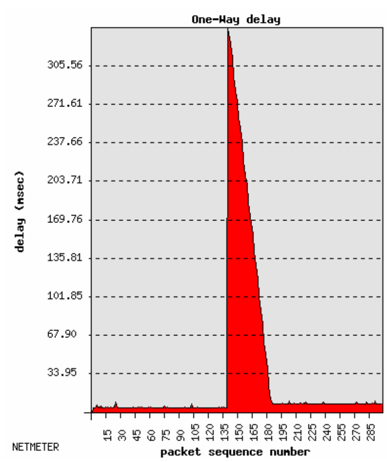


Figure 4. *FMIPv6 handover (instantaneous OWD)*

Figure 4 shows a sample for the FMIPv6 handover. We can clearly see that no packet is lost while regarding the OWD we see a spike. This behaviour is due to the protocol operational procedures. While the MN is changing its point of attachment (from one AP to the other) the old access router is tunnelling and forwarding packets to the new one and the new access router, at the same time, it is buffering packets until the MN regains connectivity. So, FMIPv6 delays (buffers) packets instead of losing them. The packets will be stored in a buffer while the MN's IEEE 802.11 layer is disconnected; hence, the packet's maximum delay is equal to the IEEE 802.11 handover latency.

4.2 Handover Latency

The whole system was tested doing a set of handovers, capturing all the signalling messages and processing them off line using "PHM Tool". Table I (results in milliseconds) shows the results of the handover latency for the different mobility protocols without taking into account the wireless part.

MIPv6 has the slower handover due to the IPv6 part. The MN has to perform *Neighbor Unreachability Detection* and *Duplicate Address Detection* to realize that its previous access router is no longer reachable and to check that its new CoA is unique on that link. Both algorithms have timeouts and they were set to their minimum value. The Mobile IPv4 standard allows tuning the implementation and, as soon as the MN realizes that a new access router is present (upon reception of an "Agent Advertisement") obtains a new CoA and forgets about the old access router. IPv4 does not perform DAD, the CoA is the FA address. The Mobile IPv6 protocol relies on "Neighbor Discovery" autoconfiguration which, due to security reasons, is difficult to speed up.

FMIPv6 has very low handover latency as expected. Note that the IEEE 802.11 part of the handover latency has not been taking into account (for any of the protocols) because it's a common part for all of them.

Table I. *Handover Latency comparison for mobility protocols (ms)*

	MIPv4		MIPv6		FMIPv6	
	Mean	**Std. Dev**	**Mean.**	**Std. Dev**	**Mean**	**Std. Dev**
Handover Latency	104.5	6.3	1848.2	440.1	83.4	5.5

4.3 Packet Losses

We compute the packet losses for the different mobility protocols using active measurements. Table II shows a summary of the obtained results detailed for the different active tests performed (different flow direction and different traffic types).

Table II. *Packet Losses comparison for mobility protocols (ms)*

	MIPv4		MIPv6		FMIPv6	
	Mean	**Std. Dev**	**Mean.**	**Std. Dev**	**Mean**	**Std. Dev**
CN→MN VoIP	22	12.46	61.71	17.54	0	0
CN→MN Data	42.9	13.64	207.21	65.90	0	0
MN→CN VoIP	10.58	1.98	65.80	9.78		
MN → CN Data	30.27	6.22	162	16.97		

During a handover, the packets are lost while the MN is changing its access point (IEEE 802.11), obtaining a new CoA and registering it. For MIPv4/MIPv6 as higher is the packet rate higher is the packet losses. In general, the packet losses for both protocols are the rate multiplied by the handover latency. Regarding FMIPv6 the results show that no packet is lost as expected. As explained above when the traffic source is the MN, there is no need to tunnel packets, just to buffer them on the MN (the FMIPv6 handover latency remains constant for both directions). That's why we did not perform these tests.

4.4 QoS Parameter Considerations

Tables III and IV summarizes all the results regarding the provided QoS level of the mobility protocols under test.

Table III. *OWD before/after the handover for Mobile IPv4 and Mobile IPv6 (ms)*

	Mobile IPv4				Mobile IPv6			
	Mean		Std.Dev		Mean		Std.Dev	
	Before	After	Before	After	Before	After	Before	After
CN→MN VoIP	5.0	6.0	0.8	1.5	9.7	7.4	9.6	1.6
CN→MN Data	24.1	12.8	14.7	1.3	109.2	14.3	108.3	3.8
MN → CN VoIP	4.5	2.0	5.0	0.2	5.9	4.6	3.3	2.8
MN → CN Data	16.2	4.0	22.0	0.2	19.0	6.8	15.6	0.5

MIPv4 and MIPv6 show higher delay (with also a higher value for the Standard Deviation) before than after the handover, especially for longer packets. These important QoS fluctuations before the handover are because of the wireless card. For both protocols the wireless card decides to switch to a new access point when it detects that the signal quality becomes poor, hence, the provided QoS is severely affected.

Table IV. *OWD before/after the handover for FMIPv6 (ms)*

	Fast Handovers			
	Mean		Std.Dev	
	Before	After	Before	After
CN→MN VoIP	2.7	5.1	1.3	1.7
CN→MN Data	6.3	7.5	2.8	3.8

FMIPv6 presents a different behaviour, it has low OWD fluctuations before and after the handover; however after it the OWD is slightly higher. In FMIPv6 the wireless card is forced (by the above layers) to switch from one AP to another one without having to wait until the signal quality becomes low, thus the OWD is not affected after and before the handover. Note that during the handover the packets are severely delayed (instead of lost). After the handover the OWD is higher because the packets must be routed to the old access router and tunnelled to the new access router, introducing an extra hop. As soon as the MN sends a "Binding Update" to its HA and CNs the traffic will be routed directly to the MN. This is not supported in our FMIPv6 implementation.

Table V. *IPDV before/after the handover for Mobile IPv4 and Mobile IPv6 (ms)*

	Mobile IPv4				Mobile IPv6			
	Mean		Std.Dev		Mean		Std.Dev	
	Before	After	Before	After	Before	After	Before	After
CN→MN VoIP	9.5	6.8	5.2	3.7	76.2	33.9	104.7	27.7
CN→MN Data	160.0	8.4	108.6	8.5	332.2	12.1	297.3	11.9
MN → CN VoIP	11.7	1.0	22.9	0.2	43.2	8.1	54.1	8.2
MN → CN Data	63.6	1.4	86.4	0.8	131.6	9.1	198.4	11.2

Table V presents the IPDV before and after the handover. IPDV confirms that in the MIPv4/MIPv6 handover packets suffer OWD variability before the handover due to wireless signal degradation. Due to the lack of space the IPDV results for FMIPv6 are not included, however they confirm our conclusions.

5. Conclusions

This paper focuses on a measurement-based comparison of the handover for different mobility protocols: Mobile IPv4, Mobile IPv6 and Fast Handovers for Mobile IPv6. The paper studies the handover using active and passive measurements. With passive measurements we compute the handover latency (the time spent during the handover). With active measurements we study the packet losses, and the OWD and IPDV before and after the handover in order to analyze the provided QoS level.

We use public available implementations of Mobile IPv4 and Mobile IPv6 for GNU/Linux. We have developed a Fast Handovers for Mobile IPv6 implementation that, as far as we know is the first publicly available implementation. We make our study in a real testbed.

The handover latency results show that the fastest protocol is Fast Handovers for Mobile IPv6 as expected. Mobile IPv4 performs better than Mobile IPv6 because it does not need to perform the *Duplicate Address Detection* and the *Neighbor Unreachability Detection* algorithms. These algorithms are used to ensure that its old access router is no longer reachable and that the new CoA is unique on that link.

For Mobile IPv4 and Mobile IPv6 the packet losses depend on the handover latency and the rate. During the 'handover latency' packets are lost. Fast Handovers for Mobile IPv6 has a zero packet loss. The protocol stores them (delays them) instead of loosing them. Our implementation behaves as expected.

Regarding the provided QoS level, with Mobile IPv4 and Mobile IPv6 there is a severe OWD/IPDV fluctuation before the handover. The wireless card changes from one access point to another when it detects that the signal quality becomes poor. However, Fast Handovers for Mobile IPv6 does not need to wait until the signal degrades, it forces the wireless card to switch from one access point to another, and hence, it does no suffer QoS fluctuations. Although with FMIPv6 the packets are delayed during the handover (as much as the IEEE 802.11 handover latency).

Finally, according to [20] and having into considerations the results, MIPv4 and FMIPv6 are acceptable for VoIP applications, however MIPv6 has too large handover latency and losses too many packets.

6. References

[1] IEEE 802.11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) (1997)

[2] Marco Liebesch, Xavier Pérez Costa and Ralph Scmitz: A MIPv6, FMIPv6 and HMIPv6 Handover Latency study: Analytical Approach *IST Mobile and Wireless Telecommunications Summit (2002)*

[3] Xavier Pérez Costa and Hannes Hartenstein: A simulation study on the performance of Mobile IPv6 in a WLAN-based cellular network *40 issue I Computer Networks: International Journal of Telecommunications Networking (2002)*

[4] 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)*

[5] N. Montavont and T. Noel: Handover Management for Mobile Nodes in IPv6 *IEEE Communications Magazine (2002)*

[6] Marc Torrent-Moreno, Xavier Pérez-Costa, Sebastià Sallent-Ribes A Performance Study of Fast Handovers for Mobile IPv6 *28th IEEE International Conference on Computer Networks*

[7] A. Cabellos-Aparicio, R. Serral-Gracià, L. Jakab, J. Domingo-Pascual, "Measurement based analysis of the handover in a WLAN MIPv6 scenario" *Passive and Active Measurements 2005, Springer, LNCS 3431, pp 203-214 (ISBN 3-540-25520-6)*

[8] Helsinki University of Technology: MIPL Mobile IPv6 for Linux (online) http://www.mobile-ipv6.org/ (2004)

[9] Dynamics - HUT Mobile IP (online) http://www.cs.hut.fi/Research/Dynamics/

[10] SAM – Advanced Mobile Services – Research Project MCyT (online) http://sam.ccaba.upc.edu

[11] Wireless Toolkit for Linux (online) http://www.hpl.hp.com/Jean_Tourrilhes/Linux/Tools.html

[12] T. Narte, E. Nordmark: Neighbor Discovery for IP version 6 (IPv6) RFC 2461 (1998)

[13] Thomson, S. and T. Narten, "IPv6 Address Autoconfiguration", RFC 2462, December 1998.

[14] R. Serral-Gracià, A. Cabellos-Aparicio, H. Julian-Bertomeu, J. Domingo-Pascual: Active measurement tool for the EuQoS Project. *IPS-MOME 2005 Workshop, Warsaw, Poland.*

[15] John Q. Walker, NetIQ Corporation: A Handbook for Successful VoIP Deployment: Network Testing, QoS and More (2002)

[16] P. McCann: Mobile IPv6 Fast Handovers for 802.11 Networks. draft-ietf-mipshop-80211fh-04.txt

[17] C. Perkins: IP Mobility Support for IPv4 RFC 3344 (2002)

[18] D. Johnson, C. Perkins and J. Arkko: IP Mobility Support for IPv6 RFC 3775 (2004)

[19] Mogul, J., D. Mills, J. Brittenson, J. Stone and U. Windl. Pulse-per-second API for Unix-like operating systems, version 1. Request for Comments RFC-2783, Internet Engineering Task Force, March 2000, 31 pp.

[20] Jori Liesenborgs, Prof. dr. W. Lamotte "Voice over IP in networked virtual environments" Universiteit Maastricht, (online) http://research.edm.luc.ac.be/jori/thesis/onlinethesis/contents.html

[21] Albert Cabellos-Aparicio, Jose Núñez-Martínez, Hector Julian Bertomeu, Loránd Jakab, René Serral-Gracià, Jordi Domingo-Pascual "Evaluation of the Fast Handover Implementation for Mobile IPv6 in a Real Testbed"

[22] Rajeev Koodli, "Fast Handovers for Mobile IPv6", draft-ietf-mipshop-fast-mipv6-03.txt (October 2004)

[23] Internet2 Consortium: OWAMP – NTP Configuration http://e2epi.internet2.edu/owamp/details.html#NTP (2004)