The 18th Annual IEEE International Symposium on Personal, Indoor and Mobile Radio Communications (PIMRC'07)

IPV6 SOFT HANDOVER APPLIED TO NETWORK MOBILITY OVER HETEROGENEOUS ACCESS NETWORKS

Hirokazu NAOE
Advanced Telecommunication Laboratory,
SHARP Corporation
Sophia Antipolis, France

Michelle WETTERWALD
Mobile Communication Department,
Institute EURECOM
Sophia Antipolis, France

Christian BONNET
Mobile Communication Department,
Institute EURECOM
Sophia Antipolis, France

ABSTRACT

Toward seamless inter-networking between cellular networks and Wireless LAN, one of the major challenges is seamless vertical handover between different access technologies. Even in heterogeneous access networks which provide combined radio coverage, users may still suffer from packet losses due to bad radio conditions during the handover. Eurecom IPv6 soft handover can reduce packet losses even in the case of vertical handover. Considering that Network Mobility (NEMO) has an advantage of hiding mobility functions from users involved, Eurecom IPv6 soft handover and NEMO are expected to provide a good combination. This paper proposes IPv6 soft handover extension for NEMO over heterogeneous access networks to achieve better performance for UDP and TCP traffic. The proposed method called NEMO-SHO uses two different interfaces simultaneously during vertical handovers, and provides good Quality of Service for the users regardless of the radio coverage quality. Using 3G and IEEE802.11b interfaces, our experiments show that NEMO-SHO can achieve better performances for UDP traffic and constant-bit-rate TCP traffic than Make-Before-Break handovers in lossy radio conditions. We find that in case of TCP traffic the performance gain of NEMO-SHO is more affected by the difference of bandwidth and transmission delay between the two links due to the larger amount of outstanding TCP segments.

I. INTRODUCTION

In recent years, many kinds of wireless access technologies have been deployed and used in combination for building mobile communication networks because one access technology can supplement the shortages of the others with different characteristics in terms of transmission bandwidth, radio coverage and so on. In those heterogeneous access networks where there is no common L2 architecture, vertical handovers between different access technologies need to be handled at IP layer or upper layers. As an IPv6 mobility management protocol, Mobile IPv6 [1] has been standardized in the Internet Engineering Task Force (IETF). Then NEMO Basic Support [2] has been standardized as an extension to Mobile IPv6, which allows a selected mobile node, called Mobile Router (MR), to provide mobility management functions to peripheral nodes, called Local Fixed Nodes (LFN), handling changes of its point of attachment to the access networks. In addition, the benefit of supporting multihoming for these protocols is described in [3].

The advantage of NEMO Basic Support is essentially that the LFNs require neither handling the mobility protocol nor having direct interfaces to the access networks, but the Quality of Service (QoS) for the LFNs is driven by the handover performance of the MR itself.

In case of a standard MR equipped with a single interface, performance degradations during handovers are hard to avoid without a help from L2 handover processing even using optimizations such as Fast Mobile IPv6 handover [4]. On the other hand, if the MR is multihomed with several interfaces in order to roam between heterogeneous access networks, seamless vertical handovers can be performed by establishing a new wireless link before breaking the current link. This manner, known as Make-Before-Break handover, can hide the effect of L2 disruption time. Thus, it is able to improve handover performances as presented in [5-7]. However, even with the Make-Before-Break handover, packet loss may occur due to bad radio conditions.

Eurecom IPv6 soft handover [8] has been proposed for Mobile IPv6 to reduce those packet losses. It requires a Mobile Node (MN) to use two wireless links simultaneously during handovers. It also introduces an agent on the path between the serving Access Router (AR) and its Home Agent (HA). During handovers, the IPv6 flows destined to the MN are duplicated at the agent, forwarded to the MN through the two links and merged at the MN. This approach succeeded to achieve better performance of UDP traffic on IEEE802.11b WLAN network [9], and it will achieve the same gain even in NEMO Basic Support. However the heterogeneity of wireless access technologies in terms of maximum throughput, L2 retransmission mechanism and transmissions delay has not been well investigated. Besides, the performances for other transport protocols than UDP need to be studied since TCP performance is affected by the change of bandwidth and latency during Make-Before-Break handover as shown in [10].

In this work, we first propose IPv6 soft handover extension for NEMO Basic Support, called NEMO-SHO. The MR used here is equipped with two different wireless interfaces and supported by the media-independent abstraction layer, which has been developed by Eurecom and is similar to the IEEE 802.21 framework. By using the two wireless links at the same time, NEMO-SHO aims to avoid performance degradation during vertical handovers. When a vertical handover is triggered, two bi-directional tunnels are established simultaneously between the serving AR and the MR via two wireless links. Then the packets tunnelled to the MR are duplicated at the AR, forwarded to the MR through the two tunnels and combined at the MR, and vice versa.

We then evaluate NEMO-SHO in a real testbed through the experiments with UDP traffic, Constant-Bit-Rate (CBR) TCP traffic and bulk TCP traffic taking into account the different characteristics of 3G and IEEE 802.11b WLAN interfaces. We show that NEMO-SHO can reduce packet losses for UDP traffic and stabilize throughput for CBR TCP traffic even under bad radio conditions.

This paper is organized as follows. In Section II, we describe the mechanism of NEMO-SHO. The implementation of NEMO-SHO and our test environment including radio configurations are explained in Section III. In section IV, we evaluate and analyze the handover performance of NEMO-SHO on our testbed conducting measurements of UDP traffic, CBR-TCP traffic and TCP bulk data transfer. Finally, Section V concludes the paper.

II. NEMO EXTENSION FOR IPv6 SOFTHANDOVER

NEMO-SHO introduces additional protocol operations and signalling into ARs and MRs as an extension to NEMO Basic Support. This extension allows us bicasting over two wireless links to reduce packet losses, and also to avoid injecting multiple copies of packets into LFNs when both of the duplicate packets are delivered successfully. Compared with the Make-Before-Break handover, the significant difference is whether the same packet is distributed to two wireless links or not at certain times. This means that users can rely on both links during a given time in the case of NEMO-SHO, whereas the Make-Before-Break handover forces users to rely on either the current link or the new link.

Fig. 1 presents the signalling flow of NEMO-SHO. A Previous AR (PAR) is referred as the associated AR prior to a vertical handover and a New AR (NAR) is referred as the AR subsequent to the handover. An MR has two different wireless interfaces namely I/F1 and I/F2 for vertical handovers. When the MR makes a vertical handover from the PAR to the NAR, the following operations are carried out.

A. Wireless link and Bi-directional Tunnels establishment

To determine a vertical handover, a link threshold, named handover threshold, is defined. As step A of Fig. 2 shows, the MR triggers a vertical handover when it detects that the link quality on the current link decays below the threshold even if the new link has less link quality than the current one. Then it establishes a new wireless link to the NAR using I/F2 while still using the current link (I/F1) simultaneously.

After establishing the new wireless link, a new care-of address (NCoA) is allocated by processing Router Advertisements (RA) from the NAR. The MR then sends a Local Binding Update (LBU) to the PAR from I/F2. The LBU is identical to a Mobile IPv6 Binding Update (BU) message, but has a new flag to request the PAR to establish two bi-directional IPv6-in-IPv6 tunnels. The LBU includes the previous CoA (PCoA) assigned to I/F1 in Home Address Option.

The PAR receives the LBU and creates binding information between the PCoA and the NCoA. Then the PAR returns a Local Binding Acknowledgement (LBA) to the MR via the NAR. Finally, two bi-directional tunnels are established between the PAR and the MR simultaneously: the first tunnel via I/F1, and the second tunnel via the NAR and I/F2.

B. Packet bicasting and combining process

After establishing the tunnels, the PAR duplicates the packets tunnelled from the HA to forward them to the MR through the two tunnels. When the packet is duplicated, a new IPv6 Destination Option is inserted right after IPv6-in-IPv6 encapsulation header. As shown in Fig. 3, this new option, called Packet Identifier Option (PIO), has a 16bits sequence number field for numbering the original packet. The sequence number is used for identifying the same copy of the original packet at the MR, and the number is incremented by one each time the PAR duplicates packets. When the MR receives the duplicate packet, it retrieves the sequence number from the PIO and then compares the number with its own Sequence Number Table that stores sequence number of packets arrived before. If the number already exists in the Table, the MR simply discards the packet. If not, it delivers the packet to further processing (MR-HA tunnelling operation as defined in [2]) without waiting for another copy of the same packet. This packet combining can avoid forwarding multiple copies of the original packets to LFNs. The same operation is done for uplink direction from the MR to the PAR.

C. Handover completion process

When the MR detects that the link quality on the new link reaches the handover threshold, it decides to complete the handover to the NAR (at step B of Fig. 2). Consequently, the MR sends a BU to the HA with the NCoA. Then all packets from the HA come into the NCoA via the NAR. After receiving a BA from the HA, the MR stops bicasting and sends a LBU to the PAR with lifetime zero for deregistration. Then the PAR immediately stops bicasting, deletes the binding information and returns a LBA to the MR.

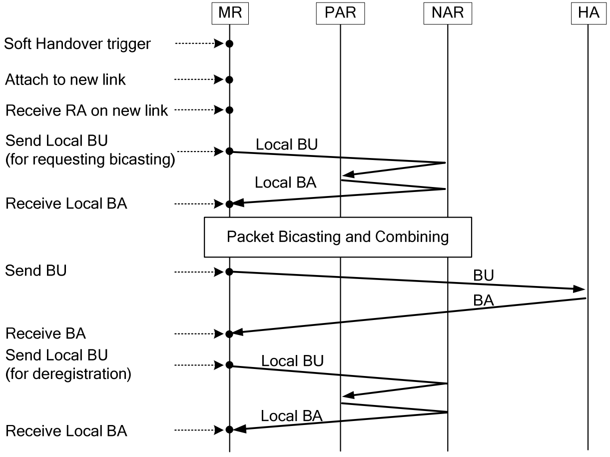

Fig. 1: NEMO-SHO signalling flow.

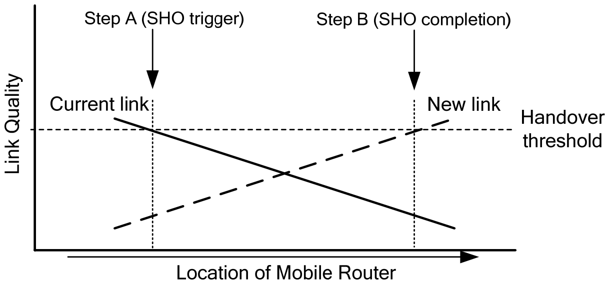

Fig. 2: Handover decision.

0	8	16	24	31
		Option Type	Option Length	
Packet Sequence Number		Reserved		

Fig. 3: Packet Identifier Option.

III. EXPERIMENT SETUP

To evaluate the performance of NEMO-SHO under the heterogeneity of wireless access technologies and different radio conditions, we carried out experiments by implementing NEMO-SHO and setting up our testbed.

A. *Implementation and testbed*

We implemented NEMO-SHO on Linux using NEPL [11] as a basis. 3G cellular interface and IEEE802.11b were selected for wireless links since these access technologies are widely deployed and have different characteristics. Regarding 3G interface, we used EURECOM wireless3g4free platform [12], which is an experimental UMTS platform providing mobile terminals with a direct IPv6 connection. In the platform, each base station creates its own IPv6 subnet and acts as a standard IPv6 router between a core network and radio access network. Fig. 4 shows the network topology of our testbed. The HA, the Correspondent Node (CN) and two ARs are connected to the same core IPv6 subnet. The 3G AR works as a PAR and acts as a base station for 3G air interface. The WLAN AR works as a NAR and acts as an IEEE802.11b Access Point (AP). Each AR is configured to send a RA to its wireless link every 1 to 3 seconds. The MR supports NEMO-SHO and has three interfaces: 3G interface and IEEE802.11b interface for vertical handovers, and 10Base-T as an ingress interface to a LFN. All nodes run on Linux kernel version 2.6.8.1.

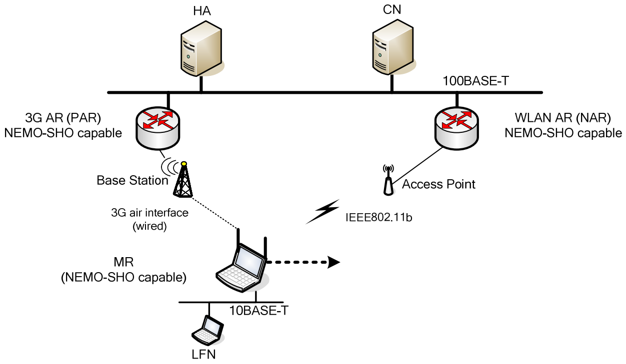

Fig. 4: Network topology of the experiment setup.

B. *Radio Configurations*

Since the experiments were conducted in our laboratory, we modified radio configurations on 3G link and WLAN link to measure handover performances under various radio conditions.

Regarding 3G link, we configured Block Error Rate (BLER) at PHY layer utilizing the Software Radio architecture of the wireless3g4free platform. 5% BLER was set as a bad radio condition (case 1) whereas nearly 0% BLER was set as a good radio condition (case 2). When setting these BLERs, a uniform random error indication of block was generated at PHY layer corresponding to the BLERs. Radio Link Control (RLC) layer used Unacknowledged Mode (UM) for UDP traffic and Acknowledged Mode (AM) for TCP traffic. The treatment of RLC Protocol Data Unit having an error indication was based on RLC specification.

As for the WLAN link, we reduced transmission power on the WLAN AR and the MR simulating the MR located away from the AP. We used minimum transmission power for a bad radio condition (case 1) whereas no modification was done for a good radio condition (case 2). Frame Error Rate (FER) was 26% and 0%, respectively when setting those power levels. Data rate was fixed to 1 Mb/s mode in order to obtain higher error resilience. Transmission retry limit was set to 1 for UDP traffic considering time-sensitivity of applications running on top of UDP. For TCP traffic, the retry limit was set to the default value on the WLAN chipset that tries RTS/CTS handshaking twice after failure of a retransmission. All the measurements were conducted on static MR, thus the radio influence due to the movement was not taken into account in this paper. All these radio configurations are summarized in Table 1.

Table 1: Radio Configuration.

Parameters		3G	WLAN
Radio Conditions	Case 1	BLER 5%	FER 26%
	Case 2	BLER 0%	FER 0%
Link Rate		384 kb/s	1 Mb/s
L2 Retransmission	UDP	RLC-UM	Retry Limit 1
	TCP	RLC-AM	2 RTS/CTS after a retry

IV. PERFORMANCE EVALUATIONS

We measured the handover performances for UDP and TCP traffic using NEMO-SHO on the testbed, and compared NEMO-SHO with the Make-Before-Break handover.

A. *Handover Performance for UDP*

UDP performance evaluation was conducted by using a traffic generator, which generates the same packet stream as VoIP traffic of Adaptive Multi-Rate Wide Band speech codec. According to Real Time Protocol payload format, given that 12.65 kb/s mode was used with bandwidth-efficient mode, the traffic generator running on the CN transmitted a 46-byte UDP payload per 20 ms to the LFN. In each radio condition, the experiment was conducted for 180 seconds, and then Packet Error Rate (PER) was calculated. To obtain PER at application layer, an acceptable latency of a packet from expected arrival time was set to 30 ms. A packet arrived behind this time is counted as a packet loss. Considering dynamic radio interferences, we activated NEMO-SHO in all trials, and then PER were measured at three points on the MR: the 3G interface, WLAN interface and the ingress interface after packet combining so that we could obtain the precise gain of NEMO-SHO.

The results of PER measurements are shown in Table 2. PER was significantly improved in case 1 by using NEMO-SHO, which verifies that NEMO-SHO can effectively reduce packet losses by using two tunnels simultaneously. If the MR starts a vertical handover using NEMO-SHO, less than 8% PER can be achieved on the VoIP application even if individual PER on the two link are more than 25%. In contrast, the ongoing VoIP session will suffer from more than 25% PER in the case of the Make-Before-Break handover. Regarding case 2, no improvement was found since the Make-Before-Break handover can achieve lossless handovers.

Table 2: PER measurements on UDP traffic.

	Case 1 (bad)		Case 2 (good)	
Make-Before-Break handover	3G	WLAN	3G	WLAN
	28.8%	26.6%	0.0%	0.0%
NEMO-SHO	8.0%		0.0%	

B. Handover Performance for TCP

Two types of TCP traffic were measured as practical use cases because actual TCP traffic patterns depend on the applications running on top of TCP. First type was bulk data transfer such as file downloading, and second type was CBR traffic such as audio streaming over TCP, e.g., Internet radio. The main difference between them in terms of TCP behaviour is as follows. In bulk data transfer, a TCP sender tries to fully utilize the available throughput to a receiver. In case of CBR traffic, the amount of outstanding TCP segments is limited by the bit rate of the stream. Then, the important criterion for evaluation is stability of transmission, i.e., less variability of average TCP throughput per second, because applications expect TCP to carry the data in time without causing buffer underflow.

Bulk transfer traffic was generated by the traffic generator, and 64 kb/s MP3 audio stream was generated by a streaming server as CBR-TCP traffic. In addition, TCP Reno and TCP Selective Acknowledgment Option were used. The experiments were made independently in three cases: using 3G interface, using WLAN interface and using NEMO-SHO.

1) TCP bulk transfer measurements

In TCP bulk transfer measurements, the CN sent bulk TCP traffic to the LFN for 10 seconds. Table 3 shows the average TCP goodput of 50 independent trials. However, no performance improvement was demonstrated in terms of average TCP goodput. The reason for this is described later in the analysis part.

Table 3: Average TCP goodput of bulk transfer.

	Case 1 (bad)		Case 2 (good)	
Make-Before-Break handover	3G	WLAN	3G	WLAN
	177 kb/s (σ 4.5)	427 kb/s (σ 129.8)	214 kb/s (σ 1.5)	750 kb/s (σ 5.4)
NEMO-SHO	407 kb/s (σ 121.4)		708 kb/s (σ 6.5)	

2) CBR-TCP traffic measurements

In CBR-TCP traffic measurement, the audio streaming was transferred from the CN to the LFN for 60 seconds. Table 4 shows standard deviations calculated from average TCP throughput per second over 5 trials. Compared with the Make-Before-Break handover, NEMO-SHO could achieve better stability of transmission, which was almost equal to the Make-Before-Break handover under good radio conditions.

Table 4: Standard deviation of CBR-TCP Throughput.

	Case 1 (bad)		Case 2 (good)	
Make-Before-Break handover	3G	WLAN	3G	WLAN
	5.3 kb/s	45.1 kb/s	4.6 kb/s	3.1 kb/s
NEMO-SHO	4.8 kb/s		N/A	

3) Analysis

When NEMO-SHO is used for TCP traffic, a key factor in providing performance improvements is the difference of bandwidth and transmission delay between two simultaneous tunnels which interacts with TCP congestion control algorithm [13].

In TCP congestion control algorithm, Congestion Window (*cwnd*) is defined as the amount of data that a TCP sender can transmit into a network without receiving an acknowledgment (ACK). When a new TCP connection is established, the *cwnd* is initialized as one segment. According to the slow start and congestion avoidance algorithm, a sender transmits TCP segments up to the *cwnd* while adjusting it in order to fully utilize the available throughput to a receiver. When a TCP segment is lost, the sender detects the loss by either expiration of TCP Retransmission Timeout (RTO) or receiving three duplicate ACKs when the *cwnd* is 4 or more, and then retransmits the lost segment.

As explained in Section II, the packet combining process doesn’t guarantee in-sequence delivery of TCP segments because it forwards the first arrived packet without waiting for the another copy. Thus, to have a positive effect of NEMO-SHO, a TCP segment lost on the high speed link (the WLAN link) must be delivered on the low speed link (the 3G link) within a time that is defined as RTO minus delay for receiving a positive ACK or a time to receive next 3 TCP segments. Otherwise, a TCP retransmission is triggered even when the low speed link successfully delivers the TCP segment later.

On the other hand, the packet arrival delay in the low speed link may increase due to accumulated queuing delay because the same amount of duplicate packets for the high speed link was also injected into the low speed link regardless the link capability. This situation negatively affects the performance of NEMO-SHO.

With this point in mind, Fig. 5 and Fig. 6 show the packet arrival time on each link at the MR in CBR-TCP traffic and TCP bulk data traffic, respectively. The shortest transmission delay is defined as zero. In CBR-TCP traffic, TCP retransmissions were only triggered by expiration of RTO because the CN kept the amount of outstanding TCP segments less than 4. In addition, the difference of the packet arrival times didn’t increase due to the attribute of CBR traffic, and most of the time, it fell within 100 ms, which was adequately shorter time than RTO (200 ms was minimum on the testbed). Hence, a packet lost on the high speed link was recovered by the low speed link without triggering a TCP retransmission. This brought out a positive effect of NEMO-SHO.

On the other hand, in TCP bulk data transfer, TCP retransmissions were also triggered by three duplicate ACKs prior to expiration of RTO because the amount of outstanding TCP segments became 4 or more. As shown in Fig. 6, the time difference was around 40 ms at the beginning. However, the difference increased rapidly due to the reason described above, and exceeded 100 ms at 4th packet, in which the packet arrival time on the 3G link obviously exceeded the time to receive next 3 TCP segments on the WLAN link. This indicates that, if this 4th TCP segment is lost on the WLAN link, a TCP fast retransmission is executed even if the MR may have an opportunity to receive the TCP segment from the 3G link. Recovering lost packets for TCP bulk data transfer by NEMO-SHO can be expected when TCP retransmissions are triggered by expiration of RTO. However, the situation is limited at the very beginning when the time difference is still small. This resulted in no performance improvement as presented in Table 3.

Another issue is that several spurious TCP retransmissions were observed even in CBR-TCP traffic. RTO is determined from smoothed average RTT that is measured over the high speed link. Thus, RTO is sometimes set to less than actual RTT of the low speed link, which is observed especially when TCP segments are delivered with increased delay caused by continuous RLC retransmissions. In such case, if a TCP segment or an ACK is lost on the high speed link, RTO expires in spite of still having a possibility to successfully deliver the packet. A possible solution will be to introduce adaptive packet combining, which absorbs the different RTT between the two links. This requires further studies.

V. CONCLUSIONS

This paper proposed and evaluated IPv6 soft handover extension for NEMO Basic Support, named NEMO-SHO, on heterogeneous access networks. By multihoming MR with two heterogeneous wireless interfaces, NEMO-SHO reduces packet losses during vertical handovers using the two wireless links simultaneously. Our experiments using 3G and IEEE802.11b interfaces showed that the gain of NEMO-SHO could be significant compensating bad radio conditions, especially for real-time traffic such as VoIP over UDP and CBR traffic over TCP. This performance improvement provides better QoS for users than the Make-Before-Break handover. In contrast, we found that in case of TCP traffic an impact of the difference of bandwidth and transmission delay between two wireless links becomes significant to limit the performance gain of NEMO-SHO due to the larger amount of outstanding TCP segments. We also observed spurious TCP retransmissions caused by different RTT between the two links, which needs to be addressed by adaptive packet combination as a future work.

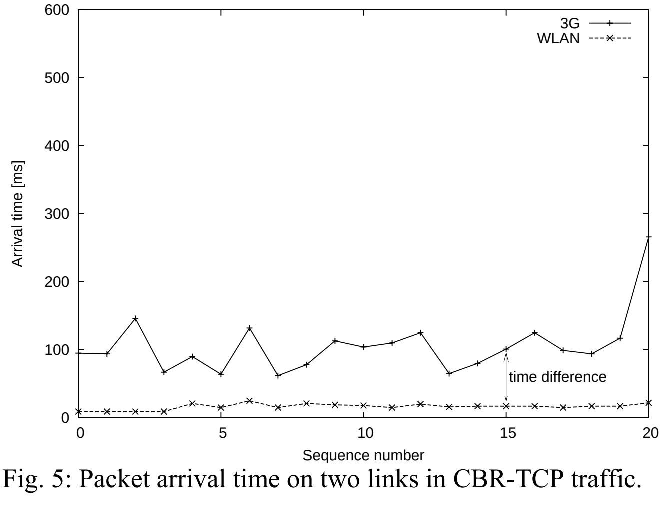

Fig. 5: Packet arrival time on two links in CBR-TCP traffic.

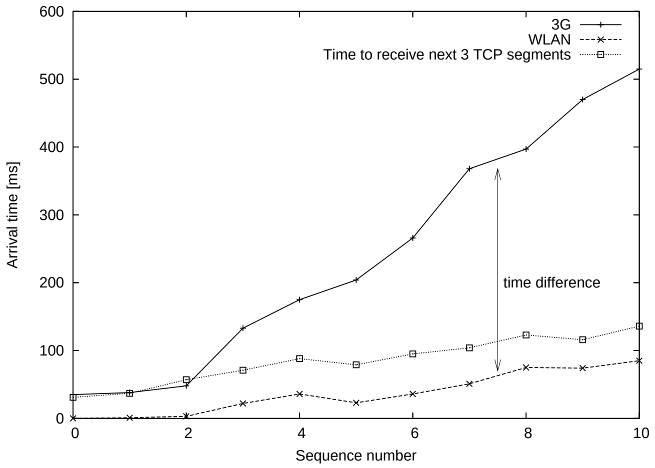

Fig. 6: Packet arrival time on two links in TCP bulk transfer.

VI. ACKNOWLEDGMENTS

We are thankful to Andreas Rüegg, Si Mohamed Zerrifi, Lionel Gauthier and Dominique Nussbaum for their help on the testbed implementation. We also appreciate the members of Advanced Telecommunication Lab. of SHARP giving an opportunity to achieve this work at Institute EURECOM.

REFERENCES

[1] David B.Johnson , Charles E.Perkins and Jari Arkko, “Mobility Support in IPv6,” IETF RFC3775, June 2004.

[2] Vijay Devarapalli, Ryuji Wakikawa, Alexandru Petrescu and Pascal Thubert, “Network Mobility (NEMO) Basic Support Protocol,” IETF RFC 3963, January 2005.

[3] T. Ernst, N. Montavont, R. Wakikawa, C. Ng, K. Kuladinithi, “Motivations and Scenarios for Using Multiple Interfaces and Global Addresses,” Internet draft, IETF, October 2006; work in progress

[4] R. Koodli, “Fast Mobile IPv6 Handover,” IETF RFC4068, July 2005.

[5] R. Chakravorty, P. Vidales, K. Subramanian, I. Pratt and J. Crowcroft, “Performance issues with vertical handovers – experiences from GPRS cellular and WLAN hot-spots integration,” *in Proceedings of IEEE Pervasive Communications and Computing Conference*, March 2004.

[6] Henrik Petander, Eranga Perera, Kun-Chan Lan and Aruna Seneviratne, “Measuring and Improving the Performance of Network Mobility Management in IPv6 Networks,” *in IEEE Journal on selected areas in communications*, September 2006

[7] Emil Lvov and Thomas Noel, “Soft Handovers over 802.11b with Multiple Interfaces,” *in IEEE Wireless Communication Systems*, September 2005.

[8] Farouk Belghoul, Yan Moret and Christian Bonnet, “Eurecom IPv6 Soft Handover,” *in Proceedings of the International Conference on Wireless Networks*, June 2003.

[9] Farouk Belghoul, Yan Moret and Christian Bonnet, “Performance comparison and analysis on MIPv6, fast MIPv6 bi-casting and Eurecom IPv6 soft handover over IEEE802.11b WLANs,” *IEEE VTC*, May 2004.

[10] Wolfgang Hansmann and Matthias Frank, “On things to happen during a TCP handover,” *in Proceedings of IEEE Local Computer Network*, October 2003.

[11] NEPL NEMO Platform for Linux, available at http://www.mobile-ipv6.org

[12] Openair interface, available at http://www.openairinterface.org

[13] M. Allman, V. Paxson, W. Stevens, “TCP Congestion Control,” IETF RFC2581, April 1999.