Analysis and Experimentation over Heterogeneous Wireless Networks

Massimo Bernaschi
Istituto per le applicazioni del calcolo
"Mauro Picone" – C.N.R.
m.bernaschi@iac.cnr.it

Filippo Cacace
Università Campus Bio-Medico di Roma
f.cacace@unicampus.it

Antonio Pescapè
Università di Napoli Federico II
pescape@unina.it

Stefano Za
Università Campus Bio-Medico di Roma
s.za@unicampus.it

Abstract

Wireless and mobile networks represent an enabling technology for ubiquitous access to information systems. However, there are critical issues that still prevent the widespread use of these technologies. In this paper we analyze and discuss our experience over a real ubiquitous network testbed capable to provide a seamless handoff among heterogeneous networks. We describe Mobile IPv6/IPv4 interoperability and an efficient mechanism, based on link-layer information, for a seamless handoff among wired and wireless networks. We present the solutions adopted in setting up a real testbed and provide an evaluation of the observed performance, including a characterization of interoperability among three wireless access network technologies: 802.11 WLAN, GPRS, and UMTS.

1. Introduction

Hospitals, airports, university campuses, large plants, represent environments where ubiquitous access to data and services (*i.e.,* ubiquitous computing) may dramatically improve the quality of service in many respects: for instance reducing the need of physical movements, improving communication among people, preventing errors and delays.

Wireless networks represent an enabling technology for ubiquitous access to information systems. However, a really ubiquitous and seamless access to a complex information infrastructure demands somehow conflicting requirements:

- access must be guaranteed when moving through different subnets, based on different technologies: Ethernet LANs, IEEE 802.11 WLANs, and 2.5/3G cellular data networks (for geographic access).
- sessions must be seamlessly maintained through different subnets. Handoff must be fast enough not to cause service degradation.
- security requirements must be enforced on all networks.
- no additional configuration effort must be required to final users.

A comprehensive solution to many of these issues is provided by IPv6 [6]. IPv6 is the next generation version of the IP protocol. The principal benefit of IPv6, and the main reason for its initial deployment, is a vastly increased address space compared with its predecessor IPv4. However IPv6 offers much more than a large address space. From the security viewpoint IPsec, part of the IPv6 specification, provides security services at the IP layer that enable a system to select security protocols, determine the algorithms to use, and put in place any cryptographic keys required. Since these services are provided at the IP layer, they can be used by any higher layer protocol (e.g. TCP, UDP, etc). This is a clear advantage with respect to other existing solutions working at transport (e.g. SSL) or application level (e.g. ssh for remote access) as it is completely transparent to the final user. IPsec is available also for IPv4 as an add-on [1]. However the deployment of IPsec for IPv4 networks is very limited since many IPv4 stacks available today do not have IPsec or support only a limited subset of it.

A similar situation there is for the support to mobility. Mobile IPv6 is defined in [2, 3] and although this is still a draft, it includes many features for streamlining mobility support that are missing in IPv4 including Stateless Address Auto-configuration and Neighbor Discovery.

Actually, mobility support in IPv6 as proposed by the Mobile IP working group, follows the design for Mobile IPv4. It retains the ideas of a home network, home agent, and the use of encapsulation to deliver packets from the home network to the mobile node's current point of attachment. While discovery of a care-of address is still required, a mobile node can configure its care-of address by using Stateless Address Autoconfiguration and Neighbor Discovery. Thus, foreign agents are not required to support mobility in IPv6. For all these reasons we think that IPv6 is the strategic solution for the ubiquitous access to information systems. However, we need to take into account that transition mechanisms are required because a number of systems/services in information systems still run plain IPv4 and support neither mobility nor security extensions.

Besides the interoperability problems, there is, at least, another issue. In the case of roaming among networks, the current MobileIPv6 specifications define a handoff procedure that, working at IP level, shows, most of the times, pretty poor performance. This represent a limitation in many practical situations. Think for instance of real time applications, where acceptable disruption times should be fairly below the second, whereas with the present techniques the handoff times can be an order of magnitude higher.

In this paper we discuss our experience about providing support for ubiquitous access to information systems through heterogeneous wireless networks. We present the solutions adopted in setting up a real Mobile IPv6 testbed and provide an evaluation of the observed performance, including a characterization of interoperability among three wireless access network technologies: 802.11 WLAN, GPRS, and UMTS.

The paper is organized as follows. In Section 2 background information useful for the following discussion is given. In Section 3 we present a solution to Mobile IPv6-IPv4 interoperability. Section 4 is devoted to analyze vertical handoff latency and to discuss a technique we proposed for handoff triggering based on information gathered at link-layer. In section 5 we present some performance characterization of TCP connections when handoff between different access networks is performed. Section 6 concludes the paper.

2. Background and related work

Mobile IPv6 [2, 3] is designed to manage mobile node movements between IPv6 networks. When a mobile node (MN) is moving through other networks a router on its home subnet, known as *home agent* (HA), keeps track of the current binding of the mobile node. When a handoff takes place, the mobile node sends messages, known as *Binding Updates* (BU), both to the home agent and to any node with which is communicating, usually indicated as *Correspondent Nodes* (CN). If the correspondent node is not Mobile IPv6-enabled, packets are sent through the home agent, by means of IPv6 tunneling[4], causing triangular routing like in Mobile IPv4 [1]. If the correspondent node understands BUs it may bypass the home agent and route its own packets directly to the mobile node by using a special option in the IPv6 Routing Header.

Several mechanisms have been proposed to enhance the handoff performance of Mobile IPv6. A few of these focus on mobility among wireless networks in limited domains like corporate buildings or university campus (horizontal micromobility, [7, 8, 9, 19, 20, 21]). A comparison among these approaches, based on simulations, is reported in [22]. An accurate analysis of Mobile IPv6 horizontal handoffs on WLAN, based on experimental data, can be found in [17]: this paper describes the relationship between link layer (L2) and network layer (L3) handoffs and shows that the contribution of L2 handoff to the overall handoff delay can be predominant, especially when there are more users in the same cell. A detailed account of L2 handoff delay can be found in [23].

Methods for minimizing handoff latency and packet loss that operate above the network layer have also been proposed [14, 24, 26, 25]. In general, solutions operating at the network layer are regarded as being more suitable since they do not violate any of the fundamental Internet design principles and because they do not require any change to the protocols at the corresponding nodes.

Another active research topics is mobility among heterogeneous networks. Although the protocols can be the same, the basic mechanism in this case is called *vertical handoff*. Peculiarities of vertical handoffs justify a separate analysis:

- horizontal handoffs are typically required when an access router becomes unavailable due to mobile hosts' movement: on the contrary, a node could be connected to an Ethernet LAN and be at the same time under the coverage of a Wireless LAN or a cellular data connection.
- a vertical handoff can be initiated for convenience, rather than connectivity, reasons.
- vertical handoff latency and packet loss are affected by network overlay: it is often possible to have loss-less handoffs by performing the configu-

ration and signaling steps on the new network before leaving the old one.

One of the first projects aiming at the integration of different wireless network technologies was *MosquitoNet* [27], which proposed an improved implementation of the Mobile IPv4 protocol to allow mobile nodes to visit foreign networks that do not provide any support for mobility. Most recent projects in this area refer to the integration of cellular geographic networks with WLANs [12, 28, 29]. Experimental measures of handoff performance for Mobile IPv6 over heterogeneous networks and its effects on TCP behavior are presented in [18]: this paper highlights how differences in network link characteristics during vertical handoffs can produce severe performance problems on TCP flows.

The flexibility provided by the presence of different network interfaces poses the question of how to configure a mobile node to transparently migrate among different network environments at different levels of the network stack. In [30] an architecture for dynamic network configuration is presented, as well as the effect of transparent and non-transparent reconfiguration on different applications. [31, 32] propose mechanisms to control the selection of the most desirable packet delivery path based on the characteristics of traffic flows. A protocol based on a rule language that explicitly determines the attribution of traffic flows to different interfaces, aimed at the implementation of *mobility policies* is proposed in [10].

The use of link layer information to improve handoff performance has been proposed under different approaches most of which focused on Mobile IPv4. Reference [13] proposes different interaction schemes between link and network layers and describes simulation results. [15] proposes the use of dedicated MAC bridges connecting different 802.11 subnets: this approach is limited to homogeneous networks, i.e. horizontal handoffs. A handoff case study between GPRS and WLAN is presented in [12]: the handoff information is gathered at link layer and transmitted to a daemon program on the application level. Link layer information is taken into account in some of the mobility protocols we already mentioned, like FHMIPv6 [19], as well as in proposals aiming at network adaptivity to support mobility, like DIRAC [16], a software based router system for wireless networks designed to facilitate the implementation and evaluation of various channel-adaptive and mobility-aware protocols.

In this paper we present experimental performance characterization of an architecture for seamless handoff and in particular we point our attention on (i) a IPv6-IPv4 interoperability mechanism in charge of allowing IPv6 clients to access IPv4 hosts; (ii) a layer 2 triggering technique to monitor the status of the interface in order to speedup the handoff without waiting the network layer triggering based on Router Advertisements; (iii) a performance characterization of interoperability between heterogeneous wireless networks.

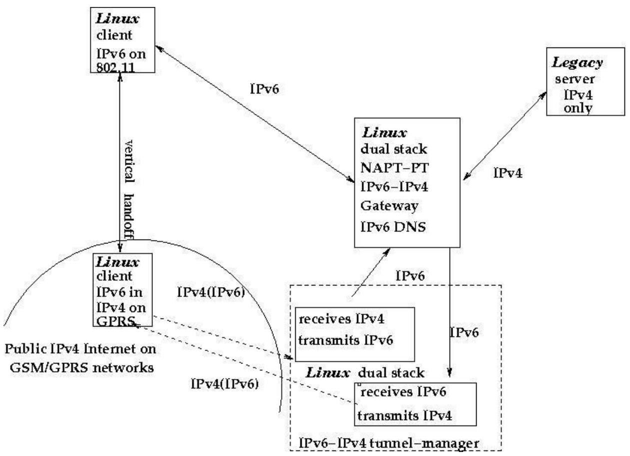

Figure 1. IPv6-IPv4 interoperability

3. Mobile IPv6-IPv4 interoperability

A number of systems/services in currently available information systems still run plain IPv4 and support neither mobility nor security extensions. To this purpose we are experimenting NAPT-PT [5] a transition mechanism that can be located at the boundary between an IPv6 and an IPv4 network. It translates IPv6 packets into IPv4 packets and vice versa. IP headers are translated and transport layer headers are modified with new port numbers. This allows transparent communication between IPv6 nodes that at the present time are the mobile clients and the IPv4 nodes that are legacy platforms providing services of typical information systems (see Fig. 1). We have extended the NAPT-PT prototype to solve the problems of connection tracking and IP fragmentation were originally not supported.

The main benefit of this approach is the simplicity: the mechanism is easy to configure, since clients needs only to use a DNS server that supports IPv6 addresses. The main drawback is that it prevents from using the IPsec component of IPv6.

A further challenge is posed by the requirement of supporting bi-directional vertical handoffs between 802.11 networks and public IP networks based on GSM/GPRS/UMTS. The problem is that most of the providers who offer IP over GSM/GPRS do not support IPv6 at this time, so there is the need of tunneling IPv6 traffic in IPv4 packets when the Mobile IPv6 clients make use of the GSM/GPRS network. Obviously IPv6 packets must be extracted so they follow the same route (through the NAPT-PT gateway) when they reach the network where hosts providing services if the information system are located. To this purpose we added one more component to the architecture: an IPv6-in-IPv4 tunnel manager (see Fig. 1), located at the Access Router, in charge of extracting the IPv6 packet when they reach the home network and encapsulate them in IPv4 packets when they leave the network.

This could appear as a tricky approach (IPv6 packets are encapsulated in IPv4 packets, then they are extracted and translated by the NAPT-PT gateway in IPv4 packets). However, we feel that support for IPv6 will be available in GSM/GPRS networks much earlier than IPv4-only networks and systems completely disappear, so we expect to keep the NAPT-PT for a long time whereas we will get rid of the tunnel manager in a (hopefully) short time. This assumption motivates the choice of using two separate components.

A detailed description of our testbed is contained in Figure 2. The experimental results reported in Section 5 have been obtained with this configuration of the testbed. IPv6 packets from the mobile node travel in the IPv6-in-IPv4 tunnel to the Access Router and then to the Home Agent. Packets who have a IPv6 destination address are sent to the corresponding node, whereas packets to IPv4 hosts, that have been marked with a special IPv6 network prefix by the mobile node, are sent to the NAPT-PT router that translates them to IPv4 packets and sends the resulting packets to the appropriate interface.

Note that in the in Figure 2 the IPv4 corresponding node is located in the home network, but the approach works also when the corresponding node is a generic IPv4 host in the Internet: in this case the NAPT-PT router sends the outgoing IPv4 packets to the Access Router. The only drawback of the NAPT-PT approach is the trangular routing resulting from the fact the the NAPT-PT mechanism must be on the path form the mobile node to the corresponding node.

The functionality of our testbed has been validated in an operating environment requiring geographic access to a hospital information system (HIS): in this case the IPv4 corresponding node is the server running the HIS. Mobile nodes can access the HIS through the wireless APs located in the hospital building, but whenever is required to access the HIS from places which are not covered by the wireless signal, the mobile node can use a VPN to connect to the Access Router using its cellular data card.

4. Vertical Handoff Performance in Mobile IPv6

The ability of using Mobile IPv6 in scenarios where communications rely on IPv4 subnetworks motivated an additional analysis to improve handoff performance in heterogeneous networks. We consider the handoff process as composed of two phases: (1) handoff detection and triggering; (2) handoff execution. A general approach to optimize handoff performance should attempt at shortening both phases [13]. Many proposals (see [7, 8, 9]) aim at reducing the second phase or, at least, reducing packet loss due to handoff execution latency. However, the detection phase is particularly important specially in vertical handoffs, where *soft* handoffs (with no packet loss) are possible.

We focused our analysis on the handoff among three representative classes of networks: (1) Ethernet LANs, characterized by high bit-rate, small power consumption and no connection cost; (2) 802.11 WLAN, with a bit-rate comparable to Ethernet LANs, but higher power consumption; (3) cellular networks (e.g. GPRS and UMTS), with lower bit-rate, high power consumption and connection cost. The ranking reflects the natural preference order among these networks. A MN would perform a *downward* handoff when a higher preference connection is lost; an *upward* move would result from the availability of a better connection. More generally, vertical handoffs can be classified as:

- *forced* handoffs, triggered by physical events regarding network interfaces availability;
- *user* handoffs, triggered by user policies and preferences.

Detection and triggering are more important for the first kind of handoff. In the second case, the MN has, most of the times, both interfaces available before starting the handoff, so simultaneous multi-access should allow handoffs with no packet loss. Handoff latency and can be further divided in the following components: (1) delay D_t for detecting lower layer events eventually leading to the handoff; (2) delay D_n for configuring an IP address on the new subnet; (3) delay D_s for handoff execution that includes the time for selecting the new router, sending signals to the HA and the CN, and latency before the arrival of packets on the new subnet. A deeper insight on the meaning of these three parameters and on their impact on handoff performance can be found in [33]. We just point out here that: (i) handoff execution always follows the first two phases, but, for vertical handoffs, phase 1 and 2 can overlap or even happen in reverse order, and (ii) current specifications of Mobile IPv6 rely on network layer mechanisms for

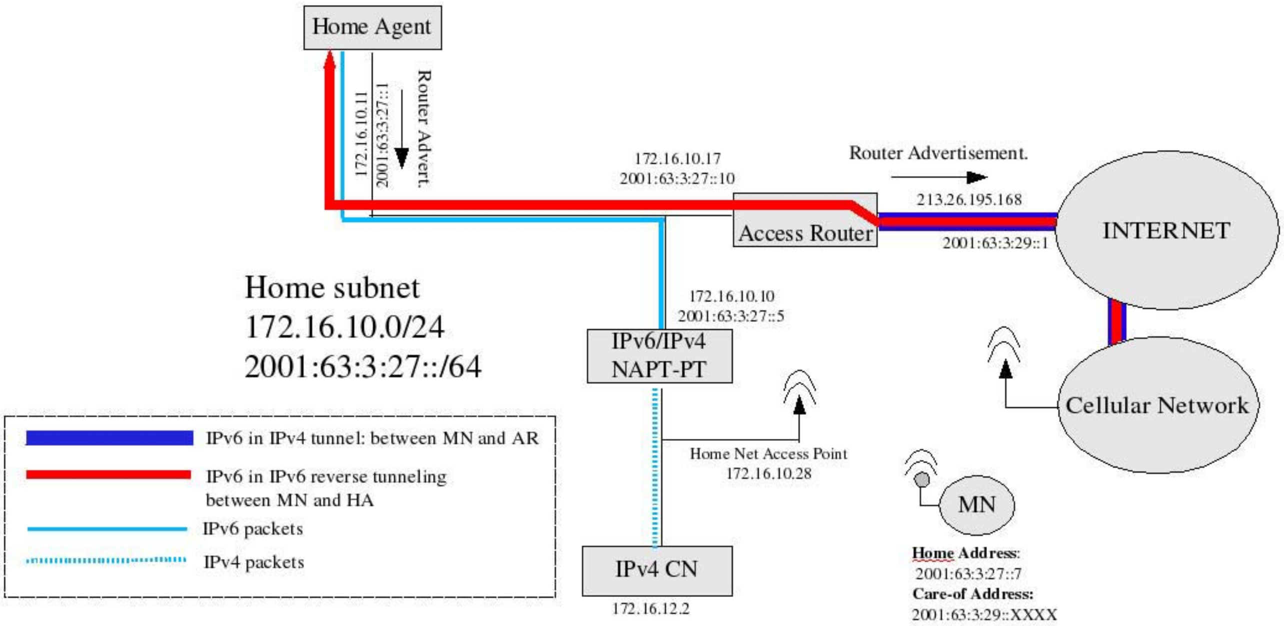

Figure 2. Experimental Testbed

detecting mobility events, although the introduction of lower layer mechanisms is suggested for implementations (see [2] section 11.5.1).

We performed experimental tests on our testbed, consisting of Linux 2.4.22 Pentium III PCs using the MIPL 1.0 Mobile IPv6 implementation [11]. Each test was repeated 10 times by using dedicated wireless network access points, with no other traffic on them. We had no access to information about the state of the public GPRS network. The Router Advertisement (RA) frequency of access routers and the Neighbor Unreachabiliy Detection (NUD) parameters used by the MIPL module, which are the factors that mainly affect handoff performance, determined, respectively, an handoff detection delay in the range of 50-1500 ms (average value is then 775 ms), and NUD times of about 500 ms for LANs and 1000 ms for GPRS and UMTS.

Experimental results led to delays of about 400 ms for user handoffs, and delays in the range 1.5-4 s for forced handoffs. The handoff triggering delay D_t has considerable impact on these times (from 47% to 98% in the case of vertical handoff from a higher preference interface with packet losses). We therefore tried to reduce triggering delay through lower layer triggering. The basic idea was to monitor the status of the interface in order to trigger the handoff without waiting the network layer triggering based on RAs. The proposed technique can be seen as a possible alternative to higher frequency RAs that are not suitable to wireless links, where they would consume the scarce bandwidth. Note that there is still an open debate in the IETF community about the employment of lower layer information at the network layer. An experimental measure of the efficiency that can be achieved by using link-layer information can help to determine whether the development of wireless systems based on lower layer triggering is worthy.

Our approach is based on the modular architecture reported in Fig. 3,

that monitors interfaces to different technologies with the aim of hiding the details of the low level interaction with the device drivers. At this time, the prototype runs in the Linux environment and is able to manage Ethernet, IEEE 802.11b, and GPRS/UMTS interfaces. The architecture can be easily extended by adding handlers for other network interfaces. The interested reader may find more details in [33].

Note that different policies can be enforced by the *Event_Handler*: A policy whose aim is to obtain seamless connectivity may keep active and configured all the network interfaces in order to minimize handoff latency at the cost of a greater power consumption, whereas a power saving policy may activate wireless interfaces only when needed. Events can regard either link availability/failure (e.g., the disconnection of an Ethernet cable or the presence of an AP) or link quality (for

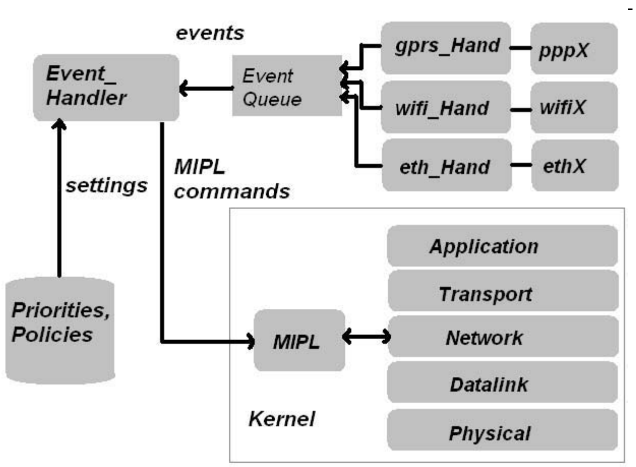

Figure 3. Proposed software architecture for lower layer triggering

forced handoffs	D_t(L3 triggering)	D_t(L2 triggering)
lan/wlan	1410 ± 390ms	200 ± 20ms
wlan/gprs	1820 ± 450ms	120 ± 50ms

Table 1. Comparison of experimental delays between network level and lower level handoff triggering

wireless links). Potential parameters for link quality are specific to the network technology and include signal strength, signal-to-interference ratio (SIR), bit error rate and frame error rate (see [13] for a survey).

The efficiency of lower layer triggering depends on fast signaling to the *Event_Handler* from the network interface handlers: these get information about the interface status with a frequency (currently 20 times per second) defined at start-up time. Table 1 shows the results of our tests: the delay shown in the table is D_t, since D_n and D_s do not change. Higher values for the frequency of interface status control would yield smaller values of the triggering delay (the response is roughly linear).

The efficiency and flexibility of an approach based on Mobile IPv6, vertical handoffs and L2 triggering can be highlighted through a comparison with the specialized protocols cited in Section 2 designed to improve Mobile IPv6, like FMIPv6 [19]. We consider the case of real-time applications on mobile hosts connected to wireless networks: in this case, the emphasis is on short handoff delays when moving through different subnets. The drawback of FMIPv6, shared by any network layer protocol, is that the total disruption time depends also on L2 handoff delay that is highly dependent on the number of clients of the visited WLAN: in [17] the total handoff delay using FMIPV6 on 802.11b networks is reported to be 152 ms with a single user and 7000 ms (worst case) with 6 users. With our approach, one could either:

- use one wireless NIC, obtaining roughly the same handoff performance (total delay is L2 delay plus about 120 ms) but without any modification to the corporate network (whereas FMIPv6 requires the deployment of a specialized protocol on the routers);
- or, use two wireless NICs and let them associate at two different APs. The handoff is triggered like an interface change (vertical handoff) by L2 interfaces management module. No change is required to the routers on the network. The cost of using two NICs is compensated by the following advantages: *i)* handoff with no packet loss; *ii)* no dependence on L2 handoff delay; *iii)* stable handoff delay of about 120 ms.

5. Experimental Results

It is clear that to make possible to future users to have ubiquitous access to novel media services we need to allow them to roam transparently across different networks, terminals and service technologies, in the same way today they are allowed to roam across different network operators with GSM/GPRS cellular devices. At the same time we have to ensure the needed QoS parameters when the handoff process is executed.

To provide a first characterization of the real network performance in presence of different *network conditions*, we measured some QoS parameters (delay, jitter, throughput), in a TCP environment (we used the Linux implementation, including the New Reno and SACKS extensions). In particular, in this section we focus on the perceived End-to-End performance at the application level.

Trials and experimentation are carried out by using an innovative tool for network performance evaluation that we called D-ITG (*Distributed Internet Traffic Generator*) [34]. D-ITG, configuring the PS (Packet Size) and IDT (Inter Departure Time) values, is in charge of generating the needed bitrate and thanks to the information stored at both sender and receiver sides is able to analyze the output results.

In all experiments the mobile node starts its communication using the cellular network (GPRS or UMTS), moves toward 802.11b wireless access network after 60 s (user handoff) and returns to the cellular network after 120 s (forced handoff). After 180 s the experiment terminates. The D-ITG sender was located over the HA node, whereas the D-ITG receiver was the MN node. Taking into account the nominal bitrate of the considered network technologies, we introduced the following "*traffic load*" classes: (i) low traffic load: PS = 32 *bytes* and IDT = $\frac{1}{100}$ *s*; (ii) medium traffic load: PS = 256 *bytes* and IDT = $\frac{1}{100}$ *s*; high traffic load: PS = 512 *bytes* and IDT = $\frac{1}{100}$ *s*. For each measured parameter and for each traffic class, several trials have been performed in the same operating conditions. The values reported in the following graphics represent a mean value across three test repetitions.

According to the approach presented in previous sections, we tested both L3 and L2 triggering, although for space reasons we will present only L2 triggering with UMTS as a cellular network.

Finally, experimental results are carried out over a real testbed (see Figure 2).

Representative experimental data are reported in Figures 4, 5, and 6. In all cases throughput, delay and jitter are reported for the same network condition. Figures 5 and 6 refer, respectively, to "low performance" UMTS connections (about 100 kb/s, of the same order of magnitude than GPRS), and maximum performance UMTS connections (about 380 kb/s). They also refer to different triggering mechanisms (L2 and L3, respectively).

A first observation concerns the performance of the cellular networks. In all cases, they exhibit higher and much more variable delays and jitter than WLAN. While this result was expected by a qualitative point of view, it is interesting to consider the mean and peak values of these parameters. In particular both GPRS and UMTS may experiment very high delay and jitter for short time intervals. Note that this is more evident when throughput is lower (Figures 4 and 5), and that these peaks do not seem related to handoffs, since they happen at different times and also before any handoff take place.

As to TCP performance during handoffs we distinguish user handoffs, when the handoff is toward a faster link, from forced handoffs, when the handoff is toward a slower link. In both cases, before discussing quantitative results, we briefly describe which is the expected behavior of the protocol.

When performing a handoff from cellular networks to WLAN, TCP may receive a more or less short sequence of out-of-order packets, due to the different speeds of the two networks. The effect of on TCP is twofold:

- at the sender end (the CN in our scenario), ACKs

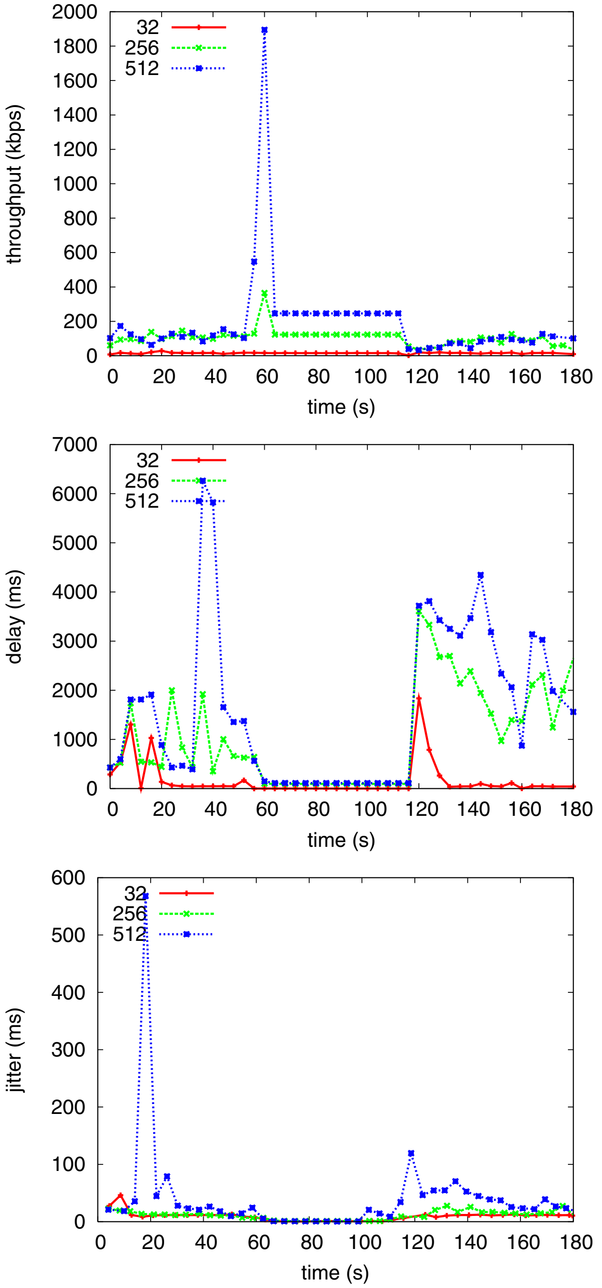

Figure 4. TCP performance (bitrate, delay and jitter) in WLAN 802.11b and GPRS scenario (L3 triggering)

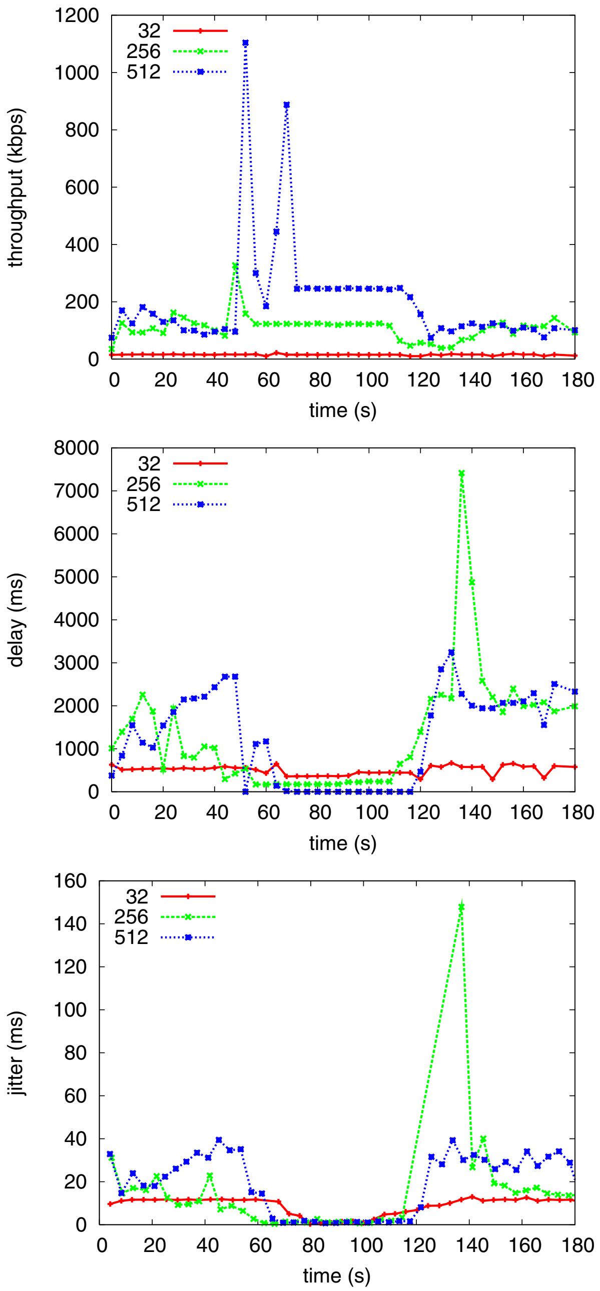

Figure 5. TCP performance (bitrate, delay and jitter) in WLAN 802.11b and UMTS scenario (L2 triggering)

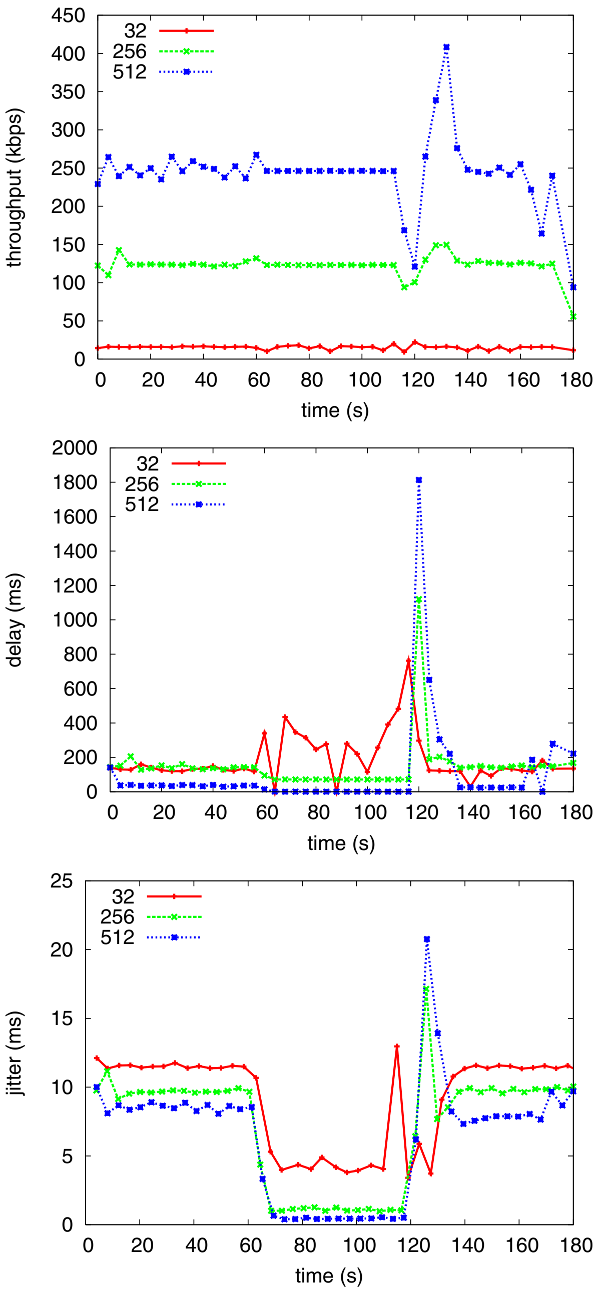

Figure 6. TCP performance (bitrate, delay and jitter) in WLAN 802.11b and UMTS scenario (L3 triggering)

sent on the WLAN arrive sooner, causing a higher transmission rate (see the peak at about 60 s in the bitrate diagrams in Figures 4 and 5);

- at the receiver end, packets sent on the WLAN arrive before previous packets sent on the cellular network. The TCP behavior depends on the version: TCP with New Reno and SACKS extensions implemented in Linux acknowledges selectively these packets, causing the re-sending of previous packets, which would in any case arrive on the cellular link.

No packets are lost, since this is a soft handoff: however a certain number of packets are retransmitted. The number of retransmitted packets depends on the traffic load. At a traffic load of 3.1 kB/s the handoff is smooth on both GPRS and UMTS links, and no packet is retransmitted. When the traffic load is 50 kB/s the average number of retransmitted packets is 27 for GPRS and 4 for UMTS. However, since packets are retransmitted on the fast WLAN link, the delay due to packet re-sending is always less than 0.2 s.

An almost seamless handoff can thus be expected at the application level. As for the perceived CN performance, however, two cases may happen. If the sustained bitrate before the handoff is limited by the cellular network performance (Figures 4 and 5), then a peak is reached at handoff for the combined effect of the early arrive of ACKs with the large amount of packets waiting for transmission. This behavior observed in our real experimentation agrees with the simulation results given in [35], where a deeper analysis of this kind of handoff is reported. Conversely, if the sustained bitrate before the handoff is not limited by the cellular network performance (Figures 6), a completely seamless handoff is performed.

In the opposite case, when a forced handoff from a fast WLAN link to a slower cellular link is performed, a certain amount of packets may be lost. These packets are not acknowledged, thus the sender retransmits the first of them with an exponential backoff interval. The negative implication of this TCP mechanism on the handoff is that, even when the Binding Update arrives at the CN causing a Binding Acknowledgment, the retransmission of lost packets is not resumed until the backoff interval expires. The average value of this additional TCP delay grows exponentially with the number of retransmission attempts, that depends on the handoff delay at the network level. The TCP delay is thus exponentially dependent on the network delay.

In these cases, the perceived CN performance varies with the performance of the handoff triggering mechanism. In all cases, after handoff the bitrate allowed by the cellular network is reached. However, while temporary remarkable drops in throughput and long transitory periods may be experienced when L3 triggering is used (Figures 4 and 6), a smoother behavior is observed when L2 triggering is used (Fig. 5). In particular, comparing the throughput graphs in Figures 4 and 5, that refer to connections with similar performance, it is apparent that the throughput drop is much more limited and that it reaches its upper bound much earlier when L2 triggering is used. Note also that even when the cellular connection is able to sustain the required throughput (Fig. 6), a longer handoff due to use of L3 triggering gives rise to higher oscillations and a longer transitory.

6. Conclusion

The use of Mobile IPv6 in real-world applications requires special attention to practical issues like the interoperability with IPv4 and the performance of the handoff procedure. We discussed how, according our experience, Mobile IPv6 effectively support mobility and showed how, by resorting to layer 2 triggering, the total handoff delay can be reduced to less than 0.2 s in case of hosts with multiple heterogeneous interfaces. The proposed approach is advantageous also from the viewpoint of simplicity and modularity since there are no additional requirements on the network stack, and it is open to be integrated with modules for the management of *mobility policies* taking into account both the user preferences and the environment constraints. We presented a performance characterization of TCP in handoff scenarios, where WLAN 802.11b, GPRS and UMTS wireless networks are involved. Experimental data about throughput, jitter, and delay indicate that, apart intrinsic performance limitations of cellular networks, almost seamless handoffs between heterogeneous wireless networks are possible with Mobile IPv6 support is used and that the resort of L2 triggering provides some additional advantages in forced handoffs.

As future work, on the one hand we plan to further characterize the performance of IPv6 mobility support with respect to UDP flows and to multimedia applications.

Acknowledgements

This work has been carried out under the financial support of the Ministero dell'Istruzione, dell'Università e della Ricerca (MIUR) in the framework of the FIRB project "Middleware for advanced services over large-scale, wired-wireless distributed systems (WEB-MINDS)”.

References

[1] C. Perkins, “IPv4 Mobility support” RFC 2002, IETF, October 1996.

[2] D. Johnson, C. Perkins and J. Arkko “Mobility Support in IPv6”, RFC 3775, IETF, June 2004.

[3] C. Perkins and D. Johnson, “Mobility Support in IPv6”, *Proc. ACM MOBICOM ’96*, pp. 27-37.

[4] A. Conta and S. Deering, “Generic Packet Tunneling in IPv6 Specification”, IETF RFC 2473.

[5] NAPTPT: http://naptpt.sourceforge.net

[6] S. Deering and R. Hinden, “Internet Protocol, version 6 (IPv6) specification”, IETF RFC2460.

[7] E. Gustavsson, A. Jonsson and C. Perkins, “Mobile IP regional registration”, Internet Draft work in progress, September 2001.

[8] A. T. Campbell, et al., “IP Micro-Mobility Protocols”,*ACM SIGMOBILE Mobile Computer and Communication Review*, Vol. 4, no. 4, October 2000, pp. 45-54.

[9] H. Soliman, C. Castelluccia, K. El-Malki, and L. Bellier, “Hierarchical MIPv6 mobility management”, Internet Draft work in progress, February 2001.

[10] J. Ylitalo, T. Jokikyyny, T. Kauppinen, A. J. Tuominen, and J. Laine, “Dynamic Network Interface Selection in Multihomed Mobile Hosts”, *Proc. IEEE HICSS ’03*, 2003.

[11] S. Kivisaari, et al., “MIPL Mobile IPv6 for Linux”, Software, http://www.mipl.mediapoli.com/ May 2000.

[12] M. Ylianttila, et al., “Handoff Procedure for Heterogeneous Wireless Networks”, *Global Telecommunications Conference Globecom 99*,December 1999, pp. 2787-2793.

[13] A. Festag, “Optimization of Handover Performance by Link Layer Triggers in IP-Based Networks: Parameters, Protocol Extensions and APIs for Implementation”,*TKN Technical Report 02-014*, Berlin, August 2002.

[14] N. Nakajima, A. Dutta, S. Das, and H. Schulzrinne, “Handoff Delay Analysis and Measurement for SIP based mobility in IPv6”, to be published, 2003.

[15] H. Yokota, A. Idoue, T. Hasegawa, and T. Kato, “Link Layer Assisted IP Fast Handoff Method over Wireless LAN Networks”, *Proc. ACM MobiCom 02*, pp. 131-139.

[16] Zerfos, P., et al. “DIRAC: A software-based Wireless Router System”, *Proc. ACM MobiCom 03*, September 2003, pp. 230- 244.

[17] Montavont, N., and T. Noel, “Analysis and Evaluation of Mobile IPv6 Handovers over Wireless LAN”, *Mobile Networks and Applications*, no.8, 2003, pp. 643-653.

[18] R. Chakravorty, P. Vidales, K. Subramanian, I. Pratt, J. Crowcroft, “Practical Experience with Wireless Networks Integration using Mobile IPv6”, MobiCom 03 Poster, *ACM Mobile Computing and Communications Review*, volume 7, n. 4, 2003.

[19] Koodli, R., “Fast Handovers for Mobile IPv6”, IETF Internet Draft work in progress, February 2004.

[20] Malki, K., and Soliman, H. “Simultaneous Bindings for Mobile IPv6 Fast Handoffs”, IETF Internet Draft work in progress, June 2002.

[21] Hsieh, R., Zhou, Z.-G., and Seneviratne, A., “S-MIP: A Seamless Handoff Architecture for Mobile IP”. *Proc. 2003 INFOCOM.*

[22] Hsieh, R., ans Seneviratne, A., “A comparison of Mechanisms for Improving Mobile IP Handoff Latency for End-to-End TCP”, *Proc.ACM MobiCom 03*, pp. 29-41.

[23] Mishra, A., M. Shin, and W. Arbaugh, “An Empirical Analysis of the IEEE 802.11 MAC Layer Handoff Process” *ACM SIGCOMM Computer Communication Review*, Volume 33, n. 2.

[24] Bakre, A., and B. R. Badrinath, “I-TCP: Indirect TCP for Mobile Hosts”, *Proc. of ICDCS*, May 1995.

[25] Riegel, M., and M. Tuxen, “Mobile SCTP”, Internet Draft work in progress, February 2003.

[26] Caceres, R., and V. N. Padmanabhan, “Fast and Scalable Handoffs in Wireless Internetworks”, *Proc. ACM MobiCom 96.*

[27] M. Baker, X. Zhao, S. Cheshire, and Jonathan Stone, “Supporting Mobility in MosquitoNet”,*Proc. USENIX 96 Annual Technical Conference.*

[28] McCann, S., and H. Flygare, “Hiperlan/2 Public Access Interworking with 3G Cellular Systems”, *Wireless Networks*, vol. 10, 2004, pp. 31-43.

[29] Luo, H., Z. Jiang, B.Kim, N.K. Shankaranarayanan, ans P. Henry, “Integrating Wireless LAN and Cellular Data for the Enterprise”, *IEEE Internet Computing*, March 2003, pp. 25-33.

[30] Inouye, J., J. Binkley, and J. Walpole, “Dynamic Network Reconfiguration Support for Mobility”, *Proc. ACM MobiCom 97*, pp. 13-22.

[31] Zhao, X., C. Castelluccia, and M. Baker, “Flexible Network Support for Mobility”, *Proc. ACM MobiCom 98*, pp. 145-156.

[32] Cheshire, S., and M. G. Baker, “Internet Mobility 4x4”, *Proc. ACM SIGCOMM 96 Conference*, pp. 318-329.

[33] Bernaschi M., Cacace F., Iannello G., “Vertical Handoff Performance in Heterogeneous Networks”, *Proc. 2004 Int. Workshop on Mobile and Wireless Networking*, Montreal, August 2004.

[34] D-ITG, Distributed Internet Traffic Generator, `http://www.grid.unina.it/software/ITG`

[35] W. Hansmann, M. Frank, M. Wolf, “Performance Analysis of TCP Handover in a Wireless/Mobile Multi-Radio Environment”, *IEEE Conference on Local Computer Networks LCN 02.*