Institute Eurécom
2229 Route des Crêtes, BP 193
F-06904 Sophia Antipolis Cédex,
France

Research report N° RR-02-071

IP-based Soft Handover in All-IP wireless networks

Farouk. Belghoul, Yan. Moret, Christian. Bonnet

September 2002

{FirstName.Name}@eurecom.fr

Abstract

The growing demand for high-speed wireless access to the Internet is the driving force behind the current trends to design All-IP wireless networks, whose radio gateway and mobile stations use IP protocol for signaling and /or data transport. That will allow a ubiquitous IP-based access by mobile users, with special emphasis on the ability to use a wide variety of wireless and wired access technologies to access the common information infrastructure. The design of an all IP wireless network requires an efficient and flexible handover management, independent of layer 2 protocol. That allows efficient mobiles stations roaming between Access routers. In this paper we propose to analyze current handover approaches in main IP-based mobility protocols in terms of complexity, efficiency, and effect on TCP performances; we discuss number of issues that motivate each handover design. A number of key design choices are identified and exploited from this analysis to present our new IPv6-based soft-handover approach.

Index Terms

Wireless mobile systems, IP based mobility, handoff, soft handover, TCP performance, Heterogeneous ALL-IP Architecture.

Contents

Table of Figures

1. Introduction

One of the most important metrics in IP-mobility protocols design is the handover performance. Handover occurs when a mobile node changes its network point of attachment from one access point to another. If not performed efficiently, handover latency, jitters and packet loss directly impact and disrupt applications quality of services. With the Internet growth and heterogeneity, it becomes crucial to design efficient pure IP based handover protocol in order to handle mobility in network layer. Many proposals were done to enhance basic mobile IPv4 [1] and mobile IPv6 handover mechanisms [3][4]. In order to minimize packets loss and data delayed-delivery in Real-time wireless application such as voice over-IP, a number of proposals such as Hierarchical Mobile IP [5], Fast Handover [9] and Bi-directional Edge Tunnel [10], are proposed with some common characteristics. Their key choice is to try to decrease delay ***D***, which is the interruption time between mobile disconnection at the old access router and its connection with network through the new point of attachment. They focused specially in that delay, ***t*,** introduced by mobility protocol, which overhead the mobile effective radio handover delay ***D*.** As illustrates in follow figure, during the second handover-step τ, the mobile tries to obtain a new address in the new network, then registers it with the mobility agent in order to become reachable at this new address.

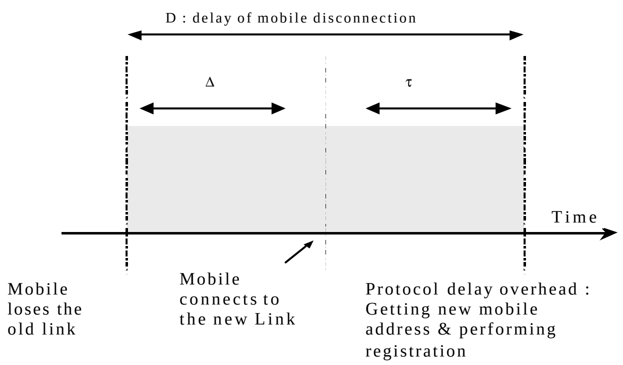

Figure 1 Handover disconnection delay

Smooth handover [7][8] uses another approach, it attacks directly Packets loss ***PL*** during the mobile disconnection and not the disconnection global delay. It introduces packets buffering mechanisms in access router to recovers lost packets while mobile disconnection. We will try to analyze this solution and its effects in packet transmission latency ***TLC***, which is taken delay to rout packets from Correspondent to Mobile before, and this before, during and after handover.

In this article we discuss the motivation behind each protocol design, present how each one applies its approach, and finally analyze their performance considering their impact in mobile disconnection latency $\boldsymbol{D}=D+t$, jitters ***TLC,*** packet losses ***PL*** and TCP connection quality. Zhang, Chen, and Agrawal [11] have identified and discussed some general design keys of an efficient IP-based soft handoff. In the current paper, we exploit those ideas to propose novel pure IPv6-based Soft Handover mechanisms, which guarantee mobility end-to-end QoS, decrease packet losses and enhance Zhang proposals, by suppressing synchronize mechanisms.

This paper is organized as follows. Section 2 contains a detailed description and comparison of the current main IP-based mobility protocols, Mobile IP, Mobile IPv6, Hierarchical, Smooth, Fast, BETH and some presented ideas about IPv4-based soft handover. We will try to analyze the handover effect in data losses and transfer delays. Handovers characteristics and their effect in TCP connections are studied in section 3, and in section 4 a new Soft Handover protocol designed for all-IP wireless network based on Ipv6 is described with Mobile-based merging mechanism for duplicated flow. Finally we discuss a number of open issues and present some concluding remarks.

2. IP-Based Mobility Landscape

This section focus on presentation and analysis of handover approaches in major IP-based mobility protocols

2.1 Mobile IPv4:

Mobile IP is the oldest and probably the most famous pure network layers solution for mobility management [1]. Its simplicity and scalability gives it a growing success. The basic principle is the uses of couple of addresses to manage Mobile node (MN) movements. Each time the MN connects to a foreign network it obtains a temporary address called Care-of-Address (CoA) in the local network, from the network Foreign Agents (FA). The MN must inform it's Home Agent (HA) of new address by the registration process, so HA will know new MN location and its corresponding FA. HA opens an IPv4 tunnel with FA, intercepts and forwards in this tunnel all IPv4 packet destined to the MN, as illustrated in "Figure 2". (We assume for figure 2 that FAs are located in Access Routers ARs). On the basis of this principle, the Mobile IP IETF working group has defined Several Internet drafts to improve Mobile IP [2].

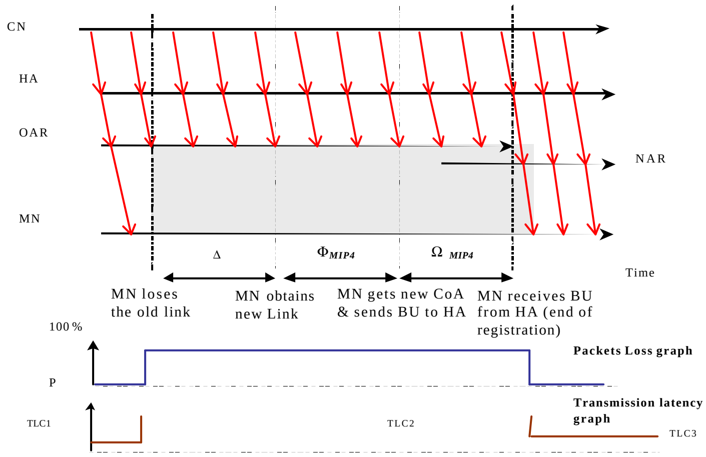

Figure 2 Data flow during MIP4 handover

Figure 2 shows us packets routing during mobile IPv4 handover. Packets sent from CN to MN are intercepted by HA, which sent them to the corresponding AR (FA), then routed to the MN. We assume D_{MIP4} the delay in handover when MN loses communication with its CN

$$D_{MIP4} = \Delta + \Phi_{MIP4} + \Omega_{MIP4}$$

D Is the laps of time between MN loses the old link with OAR, and getting new link with NAR, its independent of mobility protocol. (Φ_{MIP4}) is the time needed by MN to get new CoA from its new sub network, the delay of MN registration process, Ω_{MIP4} is Round Turn Time *RTT* between MN and HA, added to registration delay in HA. After performing all those steps, the MN is able to receive packets again through the new AR. From packet losses graph, we notice that all packets sent from CN during this process period D_{MIP4}, will be simply lost. Transmission latency graph, summarizes packets transmission latency from CN to MN, before, during and after handover:

Before, $TLC1_{MIP4} = Tr_{CN,HA} + Tr_{HA,OAR} + Radio$

During handover, $TLC2_{MIP4} = \infty$

After, $TLC3_{MIP4} = Tr_{CN,HA} + Tr_{HA,NAR} + Radio$

With, $Tr_{X,Y}$ is the packets routing delay from point X to Y in the network. ***Radio***[1] is the delay of radio transmission from AR to the MN.

[1]The radio delay is negligible compare to wireline routing delay

2.2 Mobile IPv6

Mobile IPv6 is the natural evolution of Mobile IPv4. It supports many improvements of Mobile IP and uses the advanced features of IPv6. **[3]**

In mobile IPv6, each MN is able to create quickly its own CoA using IPv6 stateless auto-configuration address mechanism, so Foreign Agents are not needed. Larger range of address is also available for mobile node in IPv6, which eliminates IPv4 address-shortage problem. In Mobile IPv6, MN can send directly binding Update to its corresponding hosts, so they can learn and cache the new mobile's CoA, and send directly packets to its NCoA without passing by HA, that is Mobile IPv6 solution to triangular routing problem.

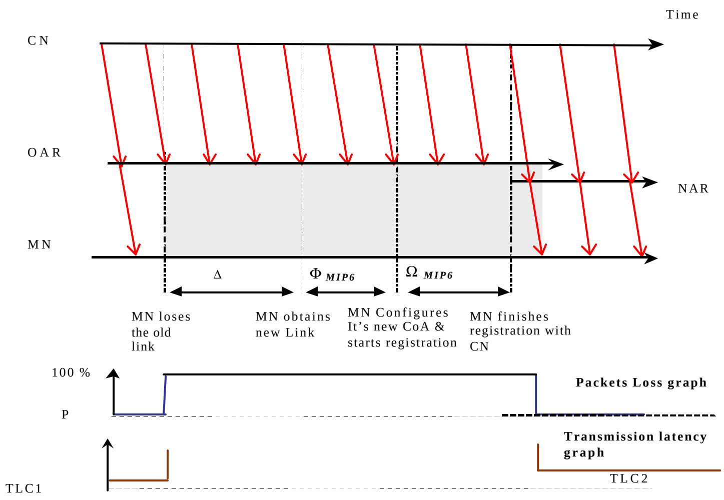

Figure 3 Mobile IPv6 handover process

From figure 3, we notice that MIP6 handover process, is identical to MIP4 one, with three consecutive steps during them the MN can't receive or sent data, with global delay:

$$D_{MIP6} = \Delta + \Phi_{MIP6} + \Omega_{MIP6}$$

D is identical in IPv4 and IPv6 protocol, it is the delay needed by MN to obtain new link. Another optimization in IPv6 is the use of Address auto configuration mechanism which reduces the step 2 delay (time needed by the mobile to obtain a new CoA: $F_{MIP6} < F_{MIP4}$). The Binding Update with CN (step3, with delay W_{MIP6} = *Round Turn Time (CN, MN)*), introduced in IPv6 registration process, reduces packets transmission latency. It allows MN to register directly its new location with its CN, so it can send packet directly to MN, without passing by MN, that make the packet transmission latency, out of MN disconnection: $TLC_{MIP6} = Tr_{CN,MN}$.

Unfortunately, Mobile IPv6 inherits from Mobile IPv4 the big disadvantage of packet loss during Handover. As we can see in packet losses graph, packets send from CN between the moment that MN leaves old network, to the end of registration process, ($D_{MIP6}=D+F_{MIP6}+W_{MIP6}$), are simply lost, without any mechanism to recover them.

2.3 Hierarchical mobile IP

The hierarchical Mobile IP protocol is a straight extension of Mobile IPv4 [4] or Mobile IPv6 [5] to support efficiently Micro-Mobility**.** This Micromobility protocol is designed for environments, where MNs change their point of attachment to the network so frequently that the basic mobile IP mechanism introduces significant network overhead and delays. It is based on the use a hierarchical architecture of Mobility Agents (MA) to handle mobile IP registration, in order to decrease signaling load over the wired IP network and registration process latency in case of local mobility.

In Hierarchical Mobile IP, network is structured in domains, each domain has its own local HA called MA, domains contains regions with local MA per region and in each region we can have multi-hierarchical sub-region, with hierarchical MA assigned to each sub region. When MN connects for the first time to a domain, it registers its CoA only one time with HA, Any further movements of MN inside sub domain generate local registration with local MA, and will be transparent for HA and sub-hierarchical MA. To perform such thing, each MA in the hierarchy must maintain an entry in its visitor's list for all MN connected to the Leaf MA at the hierarchy. Then registration process is used to establish binding between the hierarchical MA.

When MN perform a handover. Those regional registrations are only forwarded to the first MA that already has registration for this MN. The upper levels of the hierarchy are not aware of the MN movement since they don' t has to change their bindings. Data flows graph, between CN and MN during handover is globally same of Mobile IPv6 (figure2). The delay of mobile disconnection can be mapped with follow equation:

$$D_{HMIP} = \Delta + \Phi_{MIP4} + \Omega_{HMIP}$$

With, $(W_{HMIP} <= W_{MIP4})$

D is identical with MIP6 and MIP4, F is also the same. HMIP advantages can be shown in mobile movement third-step delay (between moment when MN obtains its new CoA and the end of the registration with MA): Ω_{HMIP} in HMIP is shorter in case of local mobility, because of nearest location of MA, so: $(\Omega_{HMIP} < \Omega_{MIP4})$ & $(\Omega_{HMIP} < \Omega_{MIP6})$.

Another advantage of this solution, the easy way to deploy it with the current mobile IPv6 protocol without modifications. The possibility to deploy MA at any level of hierarchy makes Hierarchical Mobile IP scalable. Regional registration allows fast moving of MN because of less registration latency, and decrease significantly signaling load in Network, also AAA can take advantages of this architecture.

However, The introduction of Mobility Agent, increases end-to-end latency because of IP packet encapsulation and tunneling. It introduces additional complexity to the architecture of wireline network [6], and the big problems still the data packet losses from the moment the MN leave the old link until the end of registration with MA or HA.

2.4 Smooth Handover

Smooth Handover [7], tries to reduce data losses during MN handover in mobile IP protocols using additional buffering mechanisms implanted in Access router.

To perform such idea, Access Routers has data memory where they buffer any received data packets before forward it to MN. When the handoff occurs, MN notifies its old Access Router address (OAR), in its registration process with the new Access Router (NAR). This one uses the given address to create a tunnel from OAR to the new one. The OAR then, forwards buffered packets to the new CoA of MN through the NAR. This basic buffer is organized as FIFO, more Optimized smooth handover buffering mechanisms was proposed in [8].

Another optimization aims to avoid that MN receives the same buffered packets before and after the handover. The way is, when mobile roams to the new network, it gives the IP headers of the last received packets from OAR to NAR, who include them on its forward-request to the OAR, This one answers by forwarding only the lost buffered packets. Hierarchical FA architecture also can be used to reduce mobile IP local handover registration overhead. Figure 4 illustrates data flows exchange between CN and MN during smooth handover, with buffered packets tunneling, from OAR to the new one at the end of registration process.

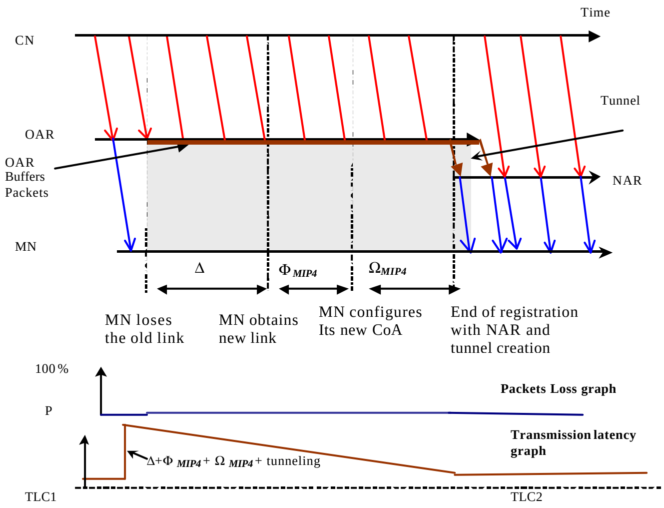

Figure 4 Smooth handover process

From Figure4 (Packet Losses graph), we can see that in theory, there is no packet loss in Smooth handover, because those misplaced packets during MN disconnection are recovered from buffers. However Smooth handoff scheme needs to introduce buffering mechanisms to each FA, which can complicates network architecture.

In reality, there is always a risk of packet losses, in case of MN takes too long time to find a new FA after lousing contact with old one, MN old FA-buffer may overflow and part of dropped packets can not be recovered at the end of handover process. Finally we can notice, that smooth handover propose solution to avoid packets lost using buffering, but it introduces additional end-to-end packets transmission latency when handover occurs (figure4, transmission latency graph). The worst case in the transmission latency is registered for packet sent at the moment in which the MN loses old link, (Figure3), before the packet arrive to MN, it should wait for:

$$TLC_{\max} = Tr_{CN,OAR} + \mathrm{A} + \Phi_{MIP4} + \Omega_{MIP4} \\ + tunneling_{(OAR,NAR)} + Radio$$

With, *Tunneling**(OAR,NAR)*** is packets routing from OAR-buffer to NAR delay.

This latency introduced by smooth handover in packets routing makes such approach not suitable for real time application.

2.5 Fast Handover

Fast Handoff uses architecture and principles of Mobile IP and tries to address a set of remaining problems, as need for a fast MN handover management for real-time applications, by reducing protocol delay overhead:

$$(t=F_{MIP}+W_{MIP}).$$

The main problem with Mobile IP in handoff management is MN movement detection, it is only achieved with Mobility mechanisms as The Agent Advertisement and Mobility Agent Advertisement Request. Fast Handoff assumes an interaction with layer 2 to anticipate the handoff, and allows the MN to perform its registration with the "new MA" by the way of the "old MA" [9]. The basic principle is that IP layer receives handoff events as "triggers" from the radio layer. When a handover is at the point to happen, in Network Control Mobility, network can determine the network prefix to which the MN will get to attach next and send it to the MN. This one forms a new CoA using Address auto- configuration mechanism and Start registration with Home Agent using its old Access Router. When the radio handoff is completed, and new link is formed, MN can perform Duplicate Address detection and start receiving and send packet directly. Before the new link former, old AR forwards packets to the MN's new CoA in order to decrease packet losses, and with hopes that packets will not arrive before the establishment of IP connectivity between MN and new sub network (figure5).

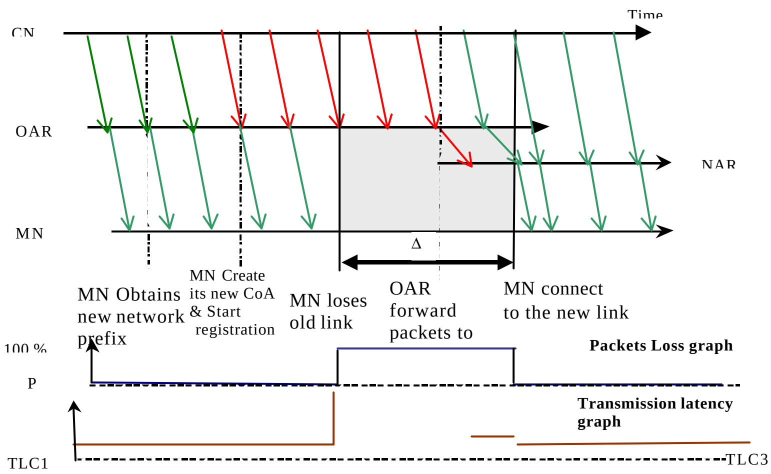

Figure 5 Fast handover process

As We see in Figure 5, Fast Handover uses layer2 interaction with IP, to predict the radio handover to suppress protocol delay overhead and to reduce mobile disconnection delay during handover process thus data packet losses. Such thing can be done only if layer2 technology allows such interaction, elsewhere; there is no any possibility to predict handover movement and its impossible then for MN to anticipate registration process. The delay between the losses of old network connection and the moment in which the MN will be ready to receive packets, $D_{fast} = D$.

We can notice there is always a chance to losses packet in two cases: 1. If MN start Radio Handover before the old AR start Forwarding Packets to the new *CoA* in the next sub network. 2. If packets forwarded to the new *CoA* arrive before the establishment of IP connection between MN and new sub network.

2.6 BETH Bi-directional Edge Tunnel Handover

As described bellow, FMIPv6 tries to reduce MIPv6 handover latency by allowing MN to anticipate the CoA assignment using L2 triggers [10]. BETH is an extension of FMIPv6, trying to eliminate the layer3 handover latency, by removing the layer 3 signaling over the air. First in BETH we define Layer 2 handover as change in link connectivity from one radio access point to another as a consequence of mobile node movement between radio coverage areas. Layer3 handover is a change in mobile IP care-of address as a consequence of mobile node movement between one sub network and another. To perform such thing, we let the MN to keep its old CoA until real time stream finished or MN slow down its movement.

When a MN detect the movement at layer 2, it can either begin the process of establishing a new CoA using standard FMIPv6 Algorithm, or it can wait until its real time stream is complete, or until its move again. If the MN kept its old CoA, a Bi-directional Edge Tunnel (BETH) is created in order to tunnel IP packets destined to MN, from the old Access Router to the new AR and to send packets from NAR to the NAR, to assure that MN traffic is routed properly. If MN decides to uses its old CoA we will have Figure6.

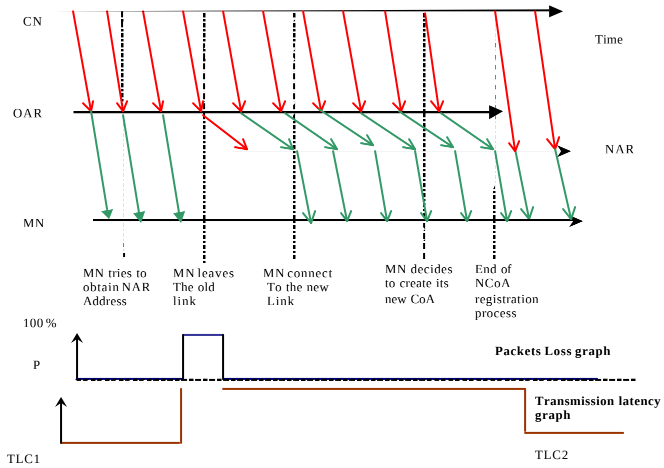

Figure 6 Handover process in BETH

We can see from Figure 6, that MN can't receive data during D1 but in the case there is no need to obtain a new CoA, it can sends and receives data immediately using its new AR after new link former (there is D2 and D3 lapse time gain). From the packet losses graph, we can see that BETH decreases packet losses compares to fast handover, but it introduces new transmission delays between OAR and NAR (transmission Latency graph). The fact that packets are tunneled from NAR to the OAR also means an overhead of typically 20 bytes will be added to each packet. **[12]** Also, the use of BETH is limited to a real time application, because in general case, when MN creates its new CoA after new link former, handover process have the same performance of Fast Mobile IP.

2.7 Handover Summary

In this chapter, we review different IP-based mobility protocols and we will focus in handover global performances based on global MN disconnection latency, transmission delay and packet losses. We assume in follows chapter that *Thp* is the average throughput of data communication between CN and MN.

Mobile IPv4 introduces basic mobility management service, it offers a pure IP layer solution for mobility support in order to continuous TCP connection even though handover causes IP addresses changes. IF we let D is the average delay needed by MN to reaches a new AR after losing the old one, F_{MIP4} the MN average delay to get new CoA from FA and W_{MIP4} the RTT between MN and HA. MN global disconnection delay when performing handover is,

$D_{MIP4} = D + F_{MIP4} + W_{MIP4}$.

All the packets sent to MN in this period are lost (packet loss = 100%.), we can map this with,

$$PL_{MIP} = Thp * (\Delta + \Phi_{MIP4} + \Omega_{MIP4})$$

Mobile IP provides also indirect and non-optimal routing mechanisms, packets sent from CN to the MN must pass first through the HA, and then routed to AR and MN by radio.

$$TLC_{MIP4} = Tr_{CN,HA} + Tr_{HA.AR} + Radio$$

That problem called triangular routing adds an addition end-to-end delay in packet transmission and adds data traffic to the network.

Mobile IPv6 integrates route optimization mechanisms such as MN Binding update with CN, to avoid triangular routing, packets are directly routed from CN to MN the end-to-end delay,

$TLC_{MIP6} = Tr_{CN,AR} + Radio$.

Enhanced IPv6 features, such as Address autoconfiguration [3] and neighbor discovery reduces time needed by MN to obtain new CoA so, $F_{MIP6} < F_{MIP4}$. The average delay of MN disconnection when handover still, $D_{MIPV6} = D + F_{MIP6} + W_{MIP6}$,

W_{MIP6} is the delay of registration with CN, $W_{MIP6} = RTT(CN,MN)$. As Mobile Ipv4, Mobile IPv6 handover generate 100% packet losses during MN disconnection, Packet losses (PL) will depends in handover delay, $PL_{MIP6} = Thp * (\Delta + \Phi_{MIP6} + \Omega_{MIP6})$

Hierarchical network architecture tries to improve mobile IP handover delays in local handover (Micro mobility), in general case handover delay, $D_{HMIP} = D + F_{MIP4} + W_{HMIP}$.

D and F are the same with basic Mobile IPv4 architecture. In case of local handover, MN sends its registration Request to its local MA (regional registration), with delay $W_{HMIP} <= W_{MIP4}$ in all cases. That makes the packet losses,

$$PL_{HMIP} = Thp * (\Delta + \Phi_{MIP4} + \Omega_{HMIP})$$

Thus, $PL_{HMIP} <= PL_{MIPV4}$

Caching mechanisms in **smooth handover** reduces the risk of packet loss, MN disconnection delay when performing handover stay the same of Mobile IP:

$$D_{Smth} = D + F_{MIP4} + W._{MIP4}$$

During this period the packets are buffered and retransmitted after to the MN through the new AR. However this buffering mechanisms introduces more end-to-end latency, if packets arrive to an AR at the moment that MN move to another links, its Transmission latency will be:

$$TLC_{\max} = Tr_{CN,OAR} + \Delta + \Phi_{MIP4} + \Omega_{MIP4}$$
$$+ tunnel_{OAR,NAR}$$

With $\boldsymbol{Tr_{CN,OAR}}$ is the delay of packets routing from CN to OAR and $tunnel_{OAR,NAR}$ presents tunneling packets delay, from OAR to the NAR. In case of packets arrived to the OAR at the end of handover process we will have:

$$TLC_{inter} = Tr_{CN,OAR} + tunnel_{(OAR \rightarrow NAR)}.$$

Otherwise $TLC = Tr_{CN,AR}$.

The packets loss during a handover can be completely eliminated in Smooth handover, unless the MN take too long to find a new network after it loses contact with its previous.

$$PL_{smth} = f(D, range\ of\ buffer).$$

End-to-end delay introduced by buffering mechanisms, make smooth handover incompatible with some sensitive applications, such as voice as voice processing and video conferencing.

Other approaches as **fast handover** and **BETH** try to reduce the global handover delays, by getting the new CoA of MN in the new link. And performing registration with HA and CN, before the radio handover. The delays between the losses of old network connection and the moment in which the MN will be ready to receive packets will be simply:

$$\boldsymbol{D_{fast} = D.}$$

And packets losses in general: $\boldsymbol{PL_{fast} = Thp * D.}$

In order to really improve quality of services of data communication during a handover, without consideration of the Wireless network layer 2 technology, Eurecom propose a pure layer3 (IP) soft handover management approach.

From the comparison bellows, we can conclude that each proposal has both strengths and weakness in handover management, which are thought to be fast and efficient, with minimal latency and packet losses.

	Data losses	Delays	Other Notice
Mobile Ipv4	MN losses data during all handover process Packet losses = 100% $\boldsymbol{PL_{MIP4}=Thp^{*}(D+F_{MIP4}+W_{MIP4})}$	$D_{MIPV4} = D+F_{MIP4}+W_{\cdot MIP4}$ $TLC_{MIP4} = Tr_{CN,HA} + Tr_{HA,AR} + Radio$	▪ Updates sent to the HA each MN move. ▪ Triangular routing
Mobile Ipv6	Data Lost from handover–start until MN obtain new CoA and finish registration with CN $PL_{MIP6}=Thp^{*}(D+F_{MIP4}+W_{MIP6})$	$D_{MIP6} = D+F_{MIP6}+W_{MIP6},$ $TLC_{MIP6} = Tr_{CN,AR}+Radio$	✓ Address auto-configure ✓ No triangular routing ✓ No need to Foreign Agent
Hierarchical	Data Lost from link left until MN obtain new CoA and finish registration with HA $PL_{HMIP} = Thp^{*}(D+F_{MIP4}+W_{HMIP})$	In local registration : better latency : $D_{HMIP} = D+F_{MIP4}+W_{HMIP}$ $(W_{HMIP4} <= W_{MIP4})$	✓ Less Handover signal load information in the network (local registration)
Smooth Handover	Data is buffered when mobile is disconnected : Low or no data losses $PL_{smth}=¦(D, buffer\ size).$	The same of IPv4: $D_{Smth} = D+F_{MIP4}+W_{MIP4}$ $TLC_{max}=Tr_{(CN,OAR)}+D+F_{MIP4}+W_{MIP4}+tunnel_{(OAR ® NAR)}$ $TLC_{inter}=tunnel_{(OAR ® NAR)}+unnel_{OAR ® NAR)}.$ $TLC = Tr_{CN,OAR}$	▪ Add buffering mechanisms to FA ▪ Not adaptive to real time Application
Fast Handover	No buffering : Possible data losses. $PL_{fast}= Thp^{*} D.$	Low latency: $D_{fast}= D.$ $TLC_{fast} = TLC_{MIP6}$	▪ Need layer 2 interaction
Beth	data losses : $PL_{beth} <= PL_{fast}$	Latency $D_{beth}=D$ $TLC_{beth}= Tr_{CN,OAR} + tunnel_{(OAR ® NAR)}$	▪ Not a general use ▪ Packets tunneling introduces data overheads.

Table 1 Summary chart of IP-based characteristics

Notice:

✓ Positive criteria

▪ Negative criteria

3. Handover Effect on TCP

Transmission control protocol TCP [14][15] is the most used transport protocol in the Internet. It provides applications with a reliable byte-oriented delivery of data on the top of IP protocol. TCP was designed and turned to perform well in wireline network, where the key functionality is to utilise the available bandwitch and avoid overloading of network.

3.1 TCP Response to Lost Packets

TCP response to lost packet is turned to work well for congestion in wired network. In the presence of high bit-errors rates and MN handover in wireless environments, packets losses is not caused always by congestion, but TCP thinks so, which causes sub-optimum performance [15].

A sent TCP packet is determined to have been lost either if no acknowledgement (*ACK*) is received within the retransmission timeout (*RTO*) period. The *RTO* is based on measurements of the round-trip delay time (*RTT*) for packets to travel over the link. These RTT measurements are collected and the *RTO* is set to the average *3* RTT*. A reasonable *RTO* is crucial to effective utilization of resources. If the *RTO* is excessive, retransmission will be unnecessarily delayed. If the *RTO* is too short, unnecessary retransmissions will occur and effective throughput will be decreased.

When a lost packet is determined by expiration of the *RTO*, TCP initiates an exponential backoff of the *RTO* and enters the slow start and congestion avoidance mode. The exponential backoff of the *RTO* involves doubling its value, with every expiration after packet retransmission. Then measures are taken to reduce the packet transmission rate so that congestion can be avoided. Slow start involves setting the congestion window, which indicates the number of packets that can be sent without causing congestion, to one packet. With each *ACK* of a received packet, the congestion window is exponentially increased. When the congestion window reaches a threshold value corresponding to half its value when the loss was determined, congestion avoidance takes over. In this phase, the congestion window is increased only linearly. When considering transmission over a wireless link, it is important to note here that multiple lost packets will cause the slow start threshold to be repeatedly reduced, and thus the congestion avoidance mode will dominate and the packet transmission rate will grow very slowly. This can lead to degradations in throughput.

3.2 TCP versions & packet losses

In the first version of TCP, if packet is not acknowledged within specific time interval RTO, the slow start procedure is entered, thus throughput degradation. In *TCP Tahoe*, after receiving a small number of duplicate acknowledgments for the same TCP packet DACK, the data sender infers that a packet has been lost and retransmits the packet without waiting for a RTO to expire, that is called fast retransmit mechanism.

Fast recovering mechanism introduced in *TCP Reno*, enhance fast retransmit to avoid slow start algorithm. The idea is that DACK also indicates that packets are leaving the network. Thus DACK can be used to clock sending of data and instead of entering slow start, Congestion window is set to half its previous value in the presence of DACK, and grows with additive increase rather than slow start. However this mechanism still only addresses single packet losses. To solve Reno TCP's performance problems when multiple packets are dropped, *TCP-SACK* uses selective acknowledgements option to enable the receiver to inform the sender about TCP packets that have been successfully received. Thus by making use of selective acknowledgements option SACK in Duplicate ACK packets, the sender can estimate which packets have been lost and retransmit them immediately with, of course, slow start algorithm avoidance.

3.3 MIPv4 & MIPv6 handovers

As we saws before, in MIP the handover delay, $D_{MIP} = D+t$, With D present movement detection delay and $t = W._{MIP}+F_{MIP}$ is the protocol delay overhead caused by MN new address configuration and registration, we can also notice that $W._{MIP} = RTT.$ During this period are lost, and if D_{MIP} is too Long *ACK* from MN will not get to CN. This one considers these unacknowledged packets to be lost and such thing has two direct negative effects on TCP performance.

First, each TCP packet has to be acknowledged before the end of $RTO=3*RTT$, if not they are retransmitted. To preve nt network congestion, during this lapse of time, D_{MIP}, the RTO is doubled for each unsuccessful transmission. This algorithm called exponential bakoff increases so much the delay of retransmission RTO beyond D_{MIP}, that at the end of registration process, there can be period of no activity in which TCP communication remain halted even after the completion of handover process [16].

Secondary, TCP will assume that packets losses during D_{MIP} is due to congestion, so mechanisms for congestion prevent as Slow Start algorithm will be used. As a result of long handover delay, TCP will repeatedly reduce its transmission-window size, and this lead to unjustified degradation in TCP throughput at the MN new AR. figure 7 show us Mobile IP handover effects on TCP connection with, $D+t=11*RTO$.

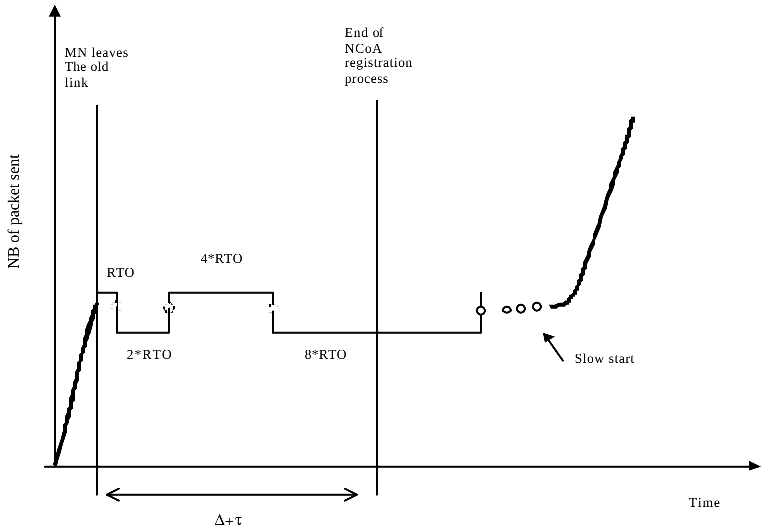

Figure 7 TCP connection during handover

3.4 Fast Handover & BETH

As described bellows, both of fast handover and BETH approaches, reduces sensibly the period of time between disconnecting from the OAR and the point in time where MN and HA are again appropriately configured to Δ.

We will base on the simulation work done in [17], to show the benefice of shorter handover delay (fast handover or BETH), on performance of modern TCP implementation employing selective acknowledgements (SACK) [18].

In long MN disconnection due of handover process, TCP sending window will fill up, which block transmission until an RTO occurs. In contrast with fast handover: 1.the sending window does not fill up during disconnection delay caused by MN handover 2.the sending window is large enough such that sender can continue send packets after the handoff is complete 3.duplicate packets indicates that packet loss occurred and require instant retransmission 4.the SACK options signal which packets to be retransmit 5.The sender will retransmit those packets 6.the slow start process can be avoid.

Let *Thp* stands for CN data rate and *S* for the size of TCP sending windows. The handover disconnection delay D_{fast}, for which the slow start algorithm will be avoid is determined via

$$Thp * (RTT + D_{fast}) < S$$

Thus,

$$D_{fast} < \frac{(S - Thp * RTT)}{Thp}$$

The work done in [18], Shows Mobile IP handover delay between 117 and 131, while fast handover leads the delay of 7 to 8 milliseconds. Using SACK option and RTT value set to 35 ms, this work shows that TCP throughput decrease In fast handover by 8%, while it is 17% in mobile IP basic handover.

3.5 Smooth handover

Smooth handover intend to reduce or suppress the number of packets dropped during MN handover process: $D_{Smth} = D + F_{MIP} + W_{MIP}$. This can be done, as we saw in 2.4, by the introduction of buffering mechanisms to recover lost packets. In follows, we will investigate TCP performances considering packet buffering mechanisms when effecting MN handover.

We consider the case where CN sends packets to the MN through an established TCP connection. First, If we let $\boldsymbol{S}$ the size of TCP sending windows, $\boldsymbol{D_{smth}}$ The MN disconnection delay, when performing handover,

$$D_{smth} = \Delta + \Phi_{MIP} + \Omega_{MIP} .$$

The requirement in Smooth handover buffer size $\boldsymbol{B_{size}}$ to recover all TCP packets dropped during this disconnection, can be determine by

$$B_{size} = \min\left(\frac{S}{RTT} * \mathrm{D}_{\mathrm{smth}}, S\right)$$

If the buffer size is smaller than $\boldsymbol{B_{size}}$,it might overflow and some TCP packets dropped early in MN handover process can not be recovered, until new *RTO* occurrence [19].

Secondary, basic Buffering mechanisms in smooth handover can not always improve TCP performance while handover, even when the size of buffer is larger then number of lost packets. In this case, if the old AR tunnels some buffered packets, which are already sent to MN successfully, towards the new AR after performing handover. Those packets, will triggers duplicate ACKs at the MN, which direct consequence is immediate activation of Fast retransmit process in CN, and the finale impact is the reduction of TCP connection rate [20].

Third, as in fast handover, the number of non-acknowledged packets is limited by the TCP window size and MN disconnection. If D_{smth} is too much bigger then RTO value, there will be

TCP packet retransmission, and if TCP sender achieves its maximum widow size, slow start algorithms will be trigged.

Worst, when MN achieves its handover process, buffered packets and retransmitted packets will triggers duplicate ACK at TCP receiver.

3.6 Conclusion

We saw that Mobile IP handover, introduces several successive timeouts that increase the RTO interval beyond the handover delay. This causes TCP connection to remain halted even after MN reconnection, and triggers slow start algorithm, which reduces TCP throughput. This one can benefit for fast handover and BETH small-handover delay and TCP selective acknowledgement. They avoid entering TCP slow start algorithm.

It was also shown that in most cases, introduction of IP-buffering mechanisms when performing handover, cannot prevent timeouts from occurring and triggering slow start algorithm, moreover TCP performance can be further degraded than the case of Mobile IP because of duplicate ACK.

4. IP-based Soft Handover

We have shown that in most general traditional IP-based handover approaches, (mobile IP, fast handover, BETH or smooth handover), it is very hard to avoid degradation of TCP performances when MN moves from OAR to NAR.

In this chapter we will try to discuss the design of non-traditional IP-based handover approach called Soft-Handover, and its effect in TCP performance.

4.1 Zhang & co propositions

In order to realize an all-IP wireless network, Tao-Zhang & All identify some efficient design keys for IP-based soft handoff [11], which means a handoff that allows a mobile station's session to progress without interruption when a Mobile Node move from one cellule to another.

These can be done, by allowing a MN to communicate with multiple Access Router simultaneously to avoid packet losses. They define two manes parts of soft handoff problem. First, multiple streams of the same IP traffic have to be distributed via multiples AR to a Mobile Node. Second, the mobile node needs some synchronization mechanisms to correctly combine data streams arrived from different radio gateway into a singles copy. To perform such thing, they propose flows-synchronization solution based on the use of Access Router to perform such thing.

To avoid that, MN uses multiple IP addresses simultaneously during soft handover and broadcasting packets to MN that are not the intended destinations, they introduce the Shadow address concepts: A unique Wirline layer 2 addressee will be assigned to MN. Each MN can be connected with multiples Access Router simultaneously, and when it starts communication with a new AR, it inserts its shadow address (x) in AR watching lists. So the new AR start accepting layer 2 frames destined to the MN Mac address, and will send carried IP packets in this frame to IP layer, in order to forward them to the Mobile Node. In Figure 8 we illustrate the Process of IPv4 soft handover, the MN is connected with Access Router1, create a second channel with AR2, send and receives packets from the two Routers, performs handover, closes first connection and use only AR2 to send and receive packets. When MN use the two Access Routers. A synchronization mechanism is needed to be sure that the IP packets received can be correctly combined.

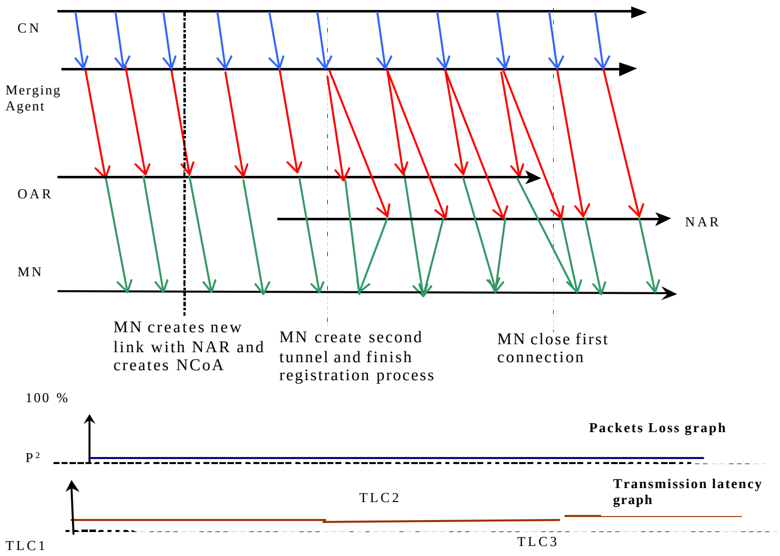

Figure 8 IPv4 based Soft Handoff

We can see from figure 8 that, this approach allows no packet losses during handoff process, which can be totally transparent, that make it very interesting for real time application. But we should notify that duplicating IP packets requires more resources in wireline and wireless network. It requires new Access Router design in order to introduce shadow address concept, which enables the distribution of multiple streams of the same IP traffic to the MN. Finally, its important to notify also that there is many open questions and problems, such as how, where and who distribute IP Packets via multiple Access routers on different IP sub nets? Which have not yet answers.

4.2 Eurecom IPv6-based Soft Handover

This paper propose a novel approach for soft handover based on IPv6, which provides additional features and new possibilities communication, such stateless auto-configuration address that facilitate the attachment of mobile node with IPv6 network

IPv6 defines also several kinds of extension headers, which may be used to perform easier handling with IPv6 packets routing, tunneling and communication securing. The defined IPv6 extension headers include destination Options headers, Hop-by-Hop Option header, Routing header and Authentication header.

Our proposed approach does not impose any change to the protocols and hardware used by Mobile IPv6. It's an extension to support an efficient Soft handover. MN can use existing radio technologies without changes.

Our soft handover is based on three Keys:

1. Data Distribution: Separate copies of the same data sent by Correspondent Node are tunneled via multiple Access Router to the same Mobile Node.
2. Handover process: The Mobile node can establish links with both old and new AR simultaneously when performing handover, it receives duplicated flows through the two AR. MN roaming from OAR to NAR can be totally transparency.
3. Merging process: The introduction of Merging Agent structure to perform data distribution and duplication from CN to MN. Merging algorithm is described to merge duplicated streams in MN, without any modification in Ipv6 address mechanisms.

4.2.1 Data Distribution

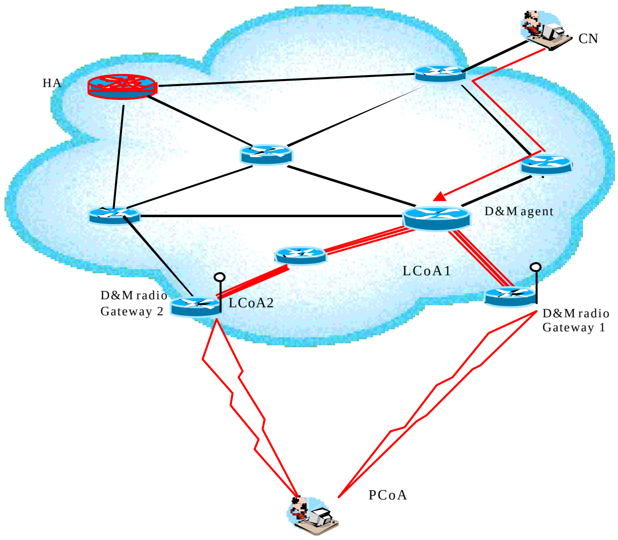

Figure 9 Data distribution

As we see in Figure 9, the proposed approach introduce a new component called "Duplication & Merging Agent". It's a conventional router located at the core network, between Correspondent node and Access Router.

D&M agent intercept packets sent to MN and duplicates them to creates two or more streams of the same data, tunnel them via multiple Access Router to the Mobile Node. The number of streams between the MN and D&M agent depends of system resource and wireless connection quality.

Consider the general case of MN with data connection with two or more Access Routers in IPv6 network, when a CN wants to send it an IP packets, the sending device will have to determine all the addresses of MN in all sub networks. To avoid that kind of problems, two substitutes Care of Address CoA (figure9) (or more if more than two different Access Routers are used simultaneously), are used for the transmission of the packets from the CN to the MN. The primary CoA (PCoA) is the temporary address (MIPv6), obtained by MN for the first time it connects to the network. It is registered within home agent and D&M agent in the reference link of MN and it's the Address used by the different correspondents, which are likely to communicate with mobile node. Each MN can have Local care-of addresses, each LCoA identify a connection of MN with an Access Router, if MN is connected with two AR, it will have first substitute CoA with first AR and second substitute CoA with second AR and both them are registered with D&M agent. To duplicates packets, the D&M agent receives packet arriving from CN and stores them in its internal memory, extracts from each packet the destination Address

(PCoA) and accesses its duplication control table to find all MN LCoA corresponding to the packet-destination PCoA. With those LCoA, D&M agent creates new IPv6 packets with the same payload information, but with substitute CoA as new destination address, Those packets are tunneled to MN via corresponding Access Router.

4.2.2 **Handover Process**

When Mobile Node MN first starts communication with an Access Router, it obtains a Primary CoA and first substitute CoA LCoA1using IPv6 address auto configuration. After this, MN can send Merging Solicitation Message MES to D&M agent, to solicit resource, if the D&M agent accept this request, initializes duplication and merging tables and respond with a merging advertisement message. MN can then perform binding update to register its PCoA with HA and CN. When the MN detect a new Access Router (AR2), it obtains another substitute CoA LcoA2 and register it with D&M agents. More than two different links for a single MN could be arranged, the limit depend only of resource availability and number of AR.

The duplication and Merging process will, as described in 4.2.2, use MN PCoA and others LCoA to intercept MN-destined IPv6 Packets, generate duplicate packets and forward them via the different Access Router.

So when MN want to moves from one AR to another, the handoff process happens in multiple steps. First the MN establish a new link with the new AR as described bellows. It can receive data from the Two Access Router simultaneously witch performs better quality of service. As the MN continues to move, eventually the signal strength from the first Access Router will be weak and not useful any longer. Again, the MN will inform the D&M agents of this fact, which will close IPv6 tunnel between D&M agent and first AR, so the MN can shut down its link with first AR and keep connected only throughout AR2. Thus, the MN transition between AR1 and AR2 will be totally transparent with no data loss and no handover delays (figure 10).

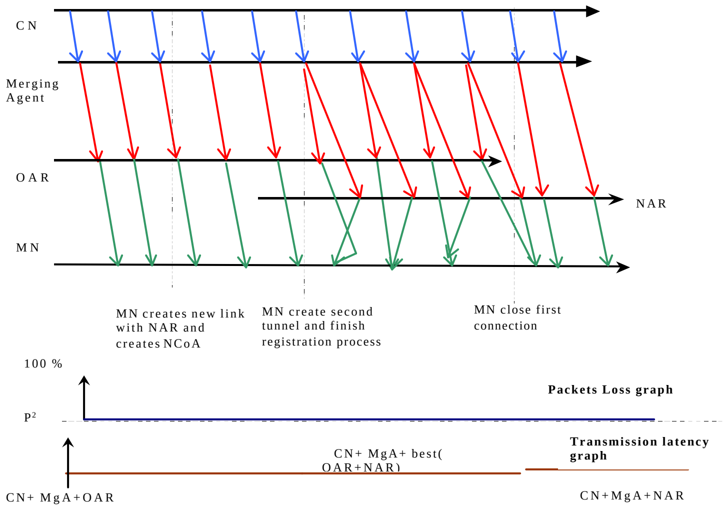

Figure 10 Eurecom IPv6-based soft handover

4.2.3 Merging process

The use of D&M agent duplication process to sent separates copies of the same data via multiple AR to the MN, can guarantee data transmission quality of service and allow MN to perform an efficient and transparent handover process without packet losses. To perform efficiently such thing, MN needs to match those multiples streams of the same data sent through multiple Access router.

When duplicating IPv6 packets, D&M agent use Destination Identifier Option (DIO) to insert merging and control information. In particular a field of the destination option defines the parameter X, which is an integer used for numbering the different packets arriving from a correspondent node, all duplicated packets from an original packets will have the same value X.

MN and D&M agent incorporate a set of table, particularly a merging control table (MCT), which defines for each entry corresponding to a list of substitute temporary CoA the values of

e corresponding to the integer which is immediately superior that the higher value of parameter X of received packet. X

X_I corresponding to packets which are been transmitted by CN, but which are not yet received in MN. Those values correspond to packets that are still missing.

In response to the reception of one packet, the process check if the DIO is included in the packet, if not, that's means that packet was not duplicated and the process route payload information to the upper layer. If the DIO is included in the packet and the source address is an entry in MCT, this means that the packet has been duplicated. Thus the value of parameter X within the DIO is read, to determine whether it's inferior to the expected value e of the MCT. If X shows to be inferior than e and the value X is not listed as missed packet in MCT, the packet is discarded. Else if X is included in the table, this means that the received packets corresponds to one packet which is still missing, the payload is routed and the value X suppressed from the table. The same thing is done if the value X shows to be superior than the expected value e, just we need also to set the new value of e to X+1, and to insert the intermediate values, between the old value of e and X, in the MCT. The merging process is described in figure11:

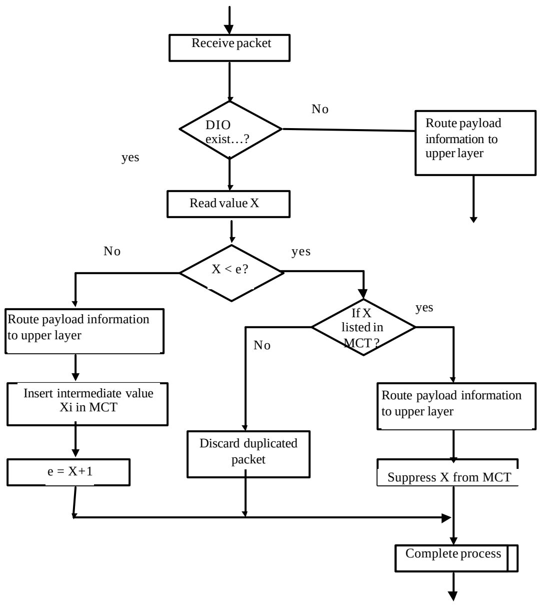

Figure 11 Merging process

4.2.4 Handover Analysis & TCP Effect

We consider two Access Router (AR1 and AR2) performing soft handover for a MN.

First step, the MN is connected to AR1, the packets transmission latency between CN and MN is:

$\boldsymbol{TLC1=Tr_{CN,D\&M}+ tunnel_{(D\&M\rightarrow AR1)} + Radio}$.

We assume P1 the probability of packet losses in this connection, global losses caused by both wireline and wireless network connection. Second step starts when MN connects to the second AR2, keeping its first connection. From this moment we will have two duplicated data flows from CN to MN through AR1 and AR2, which are merged in MN.

This merging process, routes to upper layer, the first duplicated packet arrived to MN, without consideration of its data flows, so the global packet transmission latency will be the fastest between the two flows:

$$TLC2 = Tr_{CN,D\&M} + best\ (tunnel_{(D\&M \rightarrow AR1)} + tunnel_{(D\&M \rightarrow AR1)}) + Radio.$$

If we assumes P2 the probability of packet losses in the MN connection through AR2 the probabilities of packet losses when MN is connected to the two AR (Step2) will be:

$$P = P1 * P2, \quad \text{Which is smaller than P1 or P2.}$$

Merging process in MN, merge the 2 IP flows, and filter duplicated packet at TCP receiver, so TCP layer will have only one copy of TCP packets. The results are that MN moves from AR1 to AR2 will be totally transparent for transport layer, and it will not alter the performance of TCP connection in any noticeable manner. There is no Packet loss and no duplicated packet. The third and last step in handover, occurs when MN decides to close connection with first AR, so we will have only one connection, with

$$TLC1 = Tr_{CN,D\&M} + tunnel_{(D\&M \rightarrow AR1)} + Radio.$$

This handover process is completely transparent, doesn't let any additional packet losses, and doesn't introduce end-to-end latency in packet transmission. That makes this handover protocol suitable for real time applications. Table2 summarizes Eurecom Soft handover characteristics.

	Data losses	Data Routing Delays	Notices
Soft handoff	MN Handover transparent, does not introduce additional data losses	One IP flows: **$TLC1 = Tr_{CN,D\&M} + tunnel_{(D\&M \rightarrow AR1)} + Radio$.** Duplicated IP flows **$TLC2 = Tr_{CN,D\&M} + best\ (tunnel_{(D\&M \rightarrow AR1)} + tunnel_{(D\&M \rightarrow AR1)}) + Radio.$**	▪ Requires more IP resources. ▪ Tunneling introduces overhead. ✓ Does not alter TCP connection performance

Table 2 Eurecom Soft handover characteristics.

5. Conclusion

At the end of this comparison, We can conclude that each IP-based mobility approach has both strengths and weakness. Handover management will obviously remain the most important point, which on, depend application-quality of service, the efficiency of global mobility management and TCP connection performances.

MN handover from an AR to another one must be efficient, which means minimal packets lost for general-purpose application and must be performed very fast for real- time application. If Mobile IPv4 proposes basic mobility management mechanisms with poor handover performance, which can cause unjustified degradation in TCP throughput, the IPv6 introduction enhances it, by allowing larger range of mobile addresses and fixes triangular routing problems.

Fast Handover anticipates the obtaining and the registering of future mobile address, and BETH removes layer 3 signaling over the air, which in the both case reduces global handover latency, and make them appropriate for real time applications. Combined with modern TCP implementation this handover family can avoid TCP slow start problem when MN performs handover.

Hierarchical Mobile IP aims to modifies network architecture to improve mobile IP performance in case of local mobility by restricting mobility management in local level, which improves local handover latency and decrease control information load in the network. Smooth handover is another approach, completely different, it introduces packets buffering mechanisms in access router to recovers all lost packets during handover, which make it very efficient in general application but not sweet for real time application. The size of buffer is very important to avoid degradation in TCP connections throughput.

Finally, IP-based soft handover mechanism, as presented in [11]], seems to be an efficient solution, adaptive for all kind application, with end-to-end quality of service, no MN disconnection, no packet losses, does not introduce additional delays and does not degrade TCP performance a when performing handover. The IPv6-based soft handover protocol as presented in Institute Eurecom, presents a promised pure IP solution totally transparent for transport layer. Simulation work is in progress and results will be present later.

References:

[1] Charles Perkins, editor. *“**IP mobility support for IPv4**”*. RFC 3220, January 2002.

[2] Charles Perkins,D.Johnson. IETF internet Draft ***“Optimization in Mobile IP”,*** Draft-ietf-mobileip-optim-11.txt, September 2001

[3] D. Johnson and C. Perkins. IETF Internet Draft ***"Mobility Support in IPv6",*** draft-ietf-mobileip-ipv6-12.txt. April 2000

[4] Eva Gustafsson, Annika Jonsson, Charles Perkins. IETF Internet Draft ***"Mobile IP Regional Registration",*** Mobile IP Regional Registration draft-ietf-mobileip-reg-tunnel-04.txt. March 2001

[5] Claude Castellucia. *"**A hierarchical Mobile Ipv6 Proposal**"*, 4th CTS Mobile Communication Summit, June 1999

[6] H. Soliman, C. Castellucia, K. El Malki and L. Bellier. "***Hierarchical Mobile IPv6 and Fast Handoffs***", draft-ietf-mobileip-hmipv6-00.txt, September 2000

[7] Rajeev Koodli, Charles E. Perkins. *"**A Framework for Smooth Handovers with Mobile IPv6**"*, IETF draft-koodli-mobileip-smoothv6-00.txt, 13 July 2000

[8] Charles E. Perkins and Kuang-Yeh Wang. ***"Optimized Smooth Handoffs in Mobile IP"*** Proceedings of the Fourth IEEE Symposium on Computers & Communications, July 1999

[9] Rajeev Koodli, Charles E. Perkins**.** *"**Fast Handovers in Mobile IPv6**"*, draft-koodli-mobileip-fastv6-01.txt, October 2000.

[10] James Kempf, editor ***"Bi-directional Edge Tunnel Handover for IPv6"***, IETF draft draft-kempf-beth-ipv6-02.txt, March 2001

[11] T. Zhang, J.-C. Chen, and P. Agrawal, "***Distributed soft handoff in all-IP wireless networks***", Proceedings of IEEE International Conference on Third Generation Wireless and Beyond (3Gwireless'01), (San Francisco, CA), pp. 460-465, May 2001.

[12] C. Perkins, *"**IP encapsulation within IP",*** Request for Comment (Proposed standard) 2003, Internet Engineering Task Force, Oct. 1996

[13] A.T Campbell and J.Gomez. *"**IP Micro-Mobility Protocols",*** ACM SIGMOBILE, Mobile Comp And Commun. Rev vol.4,no.4, Oct.2001, pp 45-54

[14] J .Postel. "Transmission Control Protocol" IETF RFC 793, 1981. Standard.

[15] W.R.Stevens. *"**TCP/IP Illustrated Volume 1: The Protocols"***, Addison-Wesley Longman, Reading,Massachusetts,1994.

[16] Anne Fladenmuller, Ranil De Silva. ***" The effect of Mobile IP handoffs on the performance of TCP"***, ACM Mobile Networks and Applications, 1999

[17] H. Hartenstein, K.Jonas , M. Liebsch R.Schmitz, M. Stiemerling, D. Westhoff. "***Performance of TCP in the Presence of Mobile IP Handoffs",*** (ICT 2001) IEEE International Conference on Telecom, Bucharest, Romania, June 4-7 2001

[18] M .Mathis, J. Mahdavi, Floyd, SQ Rmanow ***"TCP Selective Acknowledgement Option"***, RFC 2018, October 1996.

[19] Doo Seop Eom, Masashi Sugano, Masayuki Murata, and Hideo Miyahara. "***Performance Improvement by Packet Buffering in Mobile IP Based Networks",*** IEICE Transactions on Communications, vol. E83-B, pp. 2501-2512, November 2000.