View metadata, citation and similar papers at core.ac.uk

brought to you by CORE

provided by Repositório Institucional da Universidade de Aveiro

Universidade de Aveiro Departamento de Electrónica, Telecomunicações e Informática
2008

João Vitor Jesus Mateiro

MOBILIDADE DE EMISSORES E RECEPTORES EM REDES COM SUPORTE DE MULTICAST

Universidade de Aveiro Departamento de Electrónica, Telecomunicações e Informática
2008

João Vitor Jesus Mateiro

MOBILIDADE DE EMISSORES E RECEPTORES EM REDES COM SUPORTE DE MULTICAST

Dissertação apresentada à Universidade de Aveiro para cumprimento dos requisitos necessários à obtenção do grau de Mestre em MI em Engenharia Electrónica e Telecomunicações, realizada sob a orientação científica da Dra. Susana Sargento, Professora Auxiliar Convidada, e do Dr. Rui Aguiar, Professor Auxiliar, do Departamento de Electrónica, Telecomunicações e Informática da Universidade de Aveiro

Dedico este trabalho aos meus Pais e à minha Namorada.

O júri

Presidente

Doutor José Carlos da Silva Neves
Professor Catedrático da Universidade de Aveiro

Doutor Rui Luis Andrade Aguiar (Co-Orientador)
Professor Auxiliar da Universidade de Aveiro

Doutor Pedro Nuno Miranda Sousa
Professor Auxiliar do Departamento de Engenharia Informática da Escola de Engenharia da Universidade do Minho

Doutora Susana Isabel Barreto de Miranda Sargento (Orientadora)
Professora Auxiliar Convidada da Universidade de Aveiro

Agradecimentos

Gostaria de agradecer a várias pessoas que por diferentes razões, directa ou indirectamente, contribuíram para que esta dissertação fosse elaborada.

Primeiro que tudo, aos meus pais, que desde o início desta etapa da minha vida, me apoiaram e me deram força da forma possível.

À minha namorada, que esteve sempre do meu lado e nos momentos mais difíceis me ajudou a ultrapassá-los com o seu apoiou e carinho.

Pela sua dedicação, orientação e atenção durante a elaboração deste trabalho, à Professora Susana Sargento. Agradeço-lhe pela forma como me cativou para este trabalho e pelas suas palavras de apoio e orientação desde os primeiros instantes. A sua ajuda imprescindível para contornar problemas e encontrar soluções

Por fim, aos meus amigos e colegas. Ao Nuno Coutinho, amigo e companheiro com o qual partilhei muitas horas no estudo e desenvolvimento do NS. Ao David Vieira e Márcio Melo, igualmente bons amigo e que connosco também partilharam alguns momentos. Ao Augusto, o qual se mostrou sempre muito dedicado e cuja ajuda foi insuperável perante questões mais técnicas e relacionadas com o funcionamento do multicast em NS. A todos os outros que me foram acompanhando ao longo da minha vida e que também contribuíram para alcançar este objectivo.

Palavras-chave

Multicast, Multicast Mobility, Mobility Control, Transparency, Agent-based, Multimedia, PIM-SSM, Context Transfer Protocol, IP Multicast group, Multicast Distribution Tree.

Resumo

Este trabalho de investigação pretende apresentar, desenvolver e avaliar uma nova solução para a mobilidade de terminais em redes IP com suporte Multicast. Tendo em conta que são cada vez mais frequentes aplicações multimédia, tais como, vídeo-conferências, IPTV, entre muitas outras, as quais requerem uma elevada largura de banda e penalizam a eficácia da rede, é necessário desenvolver trabalho nesta temática a qual se encontra ainda, nos nossos dias, pouco aprofundada.

O IP Multicast consiste no envio de apenas um pacote de dados mesmo que essa informação seja pedida por vários receptores da rede. Para isso, baseia-se no conceito de grupo. Os elementos da rede, que fazem parte da chamada árvore de distribuição desse grupo multicast, replicam o pacote de forma a enviar uma cópia do mesmo sempre que o caminho para os receptores divergir.

Com o advento da tecnologia e tendo em conta as redes de próxima geração, verifica-se que estas são essencialmente baseadas no conceito de mobilidade. Por mobilidade entende-se a capacidade de um terminal se conectar a um outro elemento (Access Router) da rede.

A solução apresentada pretende oferecer de forma eficiente e transparente suporte ao movimento dos terminais minimizando os tempos de disrupção associados. Efectuou-se uma análise às soluções existentes e tendo em conta a convergência das mesmas, para resolver problemas relacionados com mobilidade de terminais multicast, uma nova solução centrada em Agent-based é apresentada.

A proposta apresentada foi testada recorrendo ao simulador NS2, demonstrando a eficiência e escalabilidade desta solução de mobilidade em redes multicast.

Keywords

Multicast, Multicast Mobility, Mobility Control, Transparency, Agent-based, Multimedia, PIM-SSM, Context Transfer Protocol, IP Multicast group, Multicast Distribution Tree.

Abstract

This research work aims to produce, develop and evaluate a new solution to the mobility of terminals in IP Multicast networks. Since, multimedia applications, such as, video-conferences, IPTV, and many others, are, nowadays, more common and because these applications require a lot of bandwidth and reduce the network efficiency, it is necessary to work on this subject which is not, yet, very developed.

IP Multicast allows sending only one data packet even if such information is requested by several receivers on the network. For that, it is based on the concept of group. The elements on the network, which are part of the multicast delivery tree for that group, replicate the packet, when the path to receivers differ, in order to send one copy for each one.

With the advent of technology and taking into account next generation networks, we see that they are essentially based on the concept of mobility. The concept of mobility means the ability of a terminal to connect to another element (Access Router) on the network.

The presented solution aims to efficiently and transparently support terminals movement minimizing the correspondent disruption time. Analyzing current solutions and taking into account their convergence, in order to solve multicast terminals mobility issues, a new Agent-based solution is presented.

The proposal was tested using the simulator NS2, demonstrating the efficiency and scalability of this mobility solution in multicast networks.

Table of Contents

Table of Figures

Tables List

List of Acronyms

A	
AP	*Access Points*
AODV	*Adhoc On-demand Distance Vector*
ASM	*Any Source Multicast*
B	
BGMP	*Border Gateway Multicast Protocol*
BT	*Bi-directional Tunnelling*
C	
CBT	*Core Based Tree*
CoA	*Care Of Address*
CTMS	*Constraint Tree Migration Scheme*
CXTP	*Context Transfer Protocol*
D	
DSDV	*Destination Sequence Distance Vector*
DVMRP	*Distance-Vector Multicast Routing Protocol*
DMSP	*Designated Multicast Service Provider*
DFA	*Domain Foreign Agent*
DSR	*Dynamic Source Routing*
DVM	*Dynamic Virtual Macrocell*
F	
FA	*Foreign Agent*
I	
IETF	*Internet Engineering Task Force*
IGMP	*Internet Group Management Protocol*
IP	*Internet Protocol*
H	
HO	*Handover*
HA	*Home Agent*
M	
MA	*Multicast Agents*
MBGP	*Multicast Border Gateway Protocol*
MIP	*Mobile IP*
MIP-BT	*Bi-directional tunnelling*
MIP-RS	*Remote subscription*
MDP	*Multicast Discover Protocol*
MLD	*Multicast Listener Discover*

MMROP	*Mobile Multicast with Routing Optimization*
MobiCast	*Multicast Scheme for Wireless Networks*
MoM	*Mobile Multicast Protocol*
MOSPF	*Multicast OSPF*
MT	*Mobile Terminal*
MTA	*Multicast Teleport Agents*
MTAMM	*MTA approach to Multicast Mobility*
MR	*Multicast Routers*
MSDA	*Multicast Source Discovery Agent*

N

NGN	*Next Generation Networks*
NS	*Network simulator*

O

OSPF	*Open Shortest Path First*

P

PIM	*Protocol Independent Multicast*
PIM – SM	*PIM – Sparse Mode*
PIM – SSM	*PIM – Source Specific Multicast*

Q

QoS	*Quality of Service*

R

RP	*Rendezvous Point*
RS	*Remote Subscription*

S

SPT	*Shortest Path Tree*
SSM	*Source Specific Multicast*

T

TCP	*Transmission Control Protocol*
TORA	*Temporally Ordered Routing Algorithm*

U

UDP	*User Datagram Protocol*

1 – Introduction

Nowadays, applications such as video/audio conference, IPTV (web), file sharing, push media and others, have an increasingly demand. In order to avoid performance degradations in the network, where data loss can be placed by packet duplication, multicast can be used. The benefits provided by mobile communications, such as commodity and portability, are increasingly attracted several users around the word. However, the bandwidth-intensive properties of multimedia applications associated with the limited capacity currently supported by wireless technologies, require restrict the bandwidth occupation. This way, the association of multicast and mobility is interesting. However, multicast is not yet very developed in mobile scenarios. Since next generations equipments are essentially mobile, some issues needs to be overcome and new solutions are welcome.

1.1 – Multicast State-of-the-Art Overview and Motivation

IP Multicast is a technique for one to many or for many to many communications over an IP infrastructure. Multicast utilizes network infrastructure efficiently by requiring the source to send each packet only once, even if it has to be delivered to a large number of listeners. Nodes in the network replicate the packet to reach multiple listeners only where necessary. Key concepts in IP Multicast include an IP Multicast group address, a multicast delivery tree and listener driven tree creation.

Mobility in IPv6 [1] is standardized in the Mobile IPv6 RFCs [2, 3]. MIPv6 [2] only roughly defines multicast mobility, using a remote subscription approach or through bi-directional tunnelling via the Home Agent. Remote subscription suffers from slow handovers, as it relies on multicast routing to adapt to the new location of terminals. Bi-directional tunnelling introduces inefficient overheads and delays due to triangular forwarding, i.e., instead of travelling on shortest paths, packets are routed through the Home Agent. Therefore, none of the approaches have been optimized for a large scale deployment. A mobile multicast service should provide a service quality compliant to real-time media distribution. However, multicast routing procedures are not easily extensible to comply with mobility requirements. Any client subscribed to a group while operating mobility handovers, requires traffic to follow to its new location; any mobile source requests the entire delivery tree to comply with or adapt to its changing positions. Significant effort has already been invested in protocol designs for mobile multicast listeners; only limited work has been dedicated to multicast source mobility, which poses the more delicate problem [4]. Multicast mobility is a generic term, which subsumes a collection of quite distinct functions. At first, multicast communication divides into Any Source Multicast (ASM) [5] and Source Specific Multicast (SSM) [6, 7]. At second, the roles of senders and listeners are distinct and asymmetric. Both may individually be mobile. Their interaction is facilitated by a multicast routing protocol such as DVMRP [8], PIM-SM/SSM [9, 10], Bi-directional PIM [11], formerly CBT [12], BGMP [13], or inter-domain multicast prefix advertisements via MBGP [14], and also, a client interaction with the multicast listener discovery protocol (MLD and MLDv2) [15, 16]. Any multicast mobility solution must take all of these functional blocks into account. It should enable seamless continuity of multicast sessions when moving from one IPv6 subnet to another. It should preserve the multicast nature of packet distribution and approximate optimal routing.

The motivation of this work is to develop a new approach that offers transparent an efficient multicast mobile terminals movements over next generation networks (NGN) to benefit many multimedia applications.

1.2 – Objectives

The purpose of this MSc. Thesis is to develop a multicast mobility solution for both listeners and senders. Nowadays, there are many multimedia applications that require data delivery from one to many hosts in the network or also from many to many. Also, equipments tend to be smaller and more portable. Furthermore, it is clear that mobility solutions can offer many advantages.

In this work, a new approach for multicast mobility named Multicast Teleport Agent approach for Multicast Mobility (MTAMM) was developed and tested. The purpose of this new solution is to allow sources and listeners movement minimizing connection loss as well as the perceptible disruption time. The main challenge was the ability to realize movements without the need to rebuild the whole multicast delivery tree. Movement is critical when a multicast source, which is sending data to a group with several listeners, aims to move.

This way, in order to implement and test the MTAMM solution, Network simulator

(NS) was used. The use of simulation allows testing many scenarios with many different characteristics. It allows evaluating the scalability of the developed work which is not always possible with real hardware since a lot of equipment is necessary and not always easy to have. Therefore, it was necessary to create and add new agents to the simulator as well as develop a new control protocol named Multicast Discover Protocol (MDP).

Taking into account that delays for real time applications are acceptable until five hundreds milliseconds (500 ms), this allows concluding through the performed evaluation that the MTAMM solution offers an efficient way to allow terminals mobility in a multicast network.

1.3 – Contributions

For the accomplishment of the above proposed objectives, this thesis presents the following set of contributions.

- The development and specification of a generic solution – Multicast Teleport Agents approach for Multicast Mobility. The base architecture and protocol (Chapter 3 – Section 3.2).
- The Architecture implementation and the realized extensions in order to develop the proposed solution (Chapter 4).
- An evaluation of the proposed solution, via simulation studies, in multiple set of scenarios and using multiple metrics (Chapter 5).

1.4 – Organization of this Thesis

This thesis consists of six chapters.

This current chapter places the thesis research in the context of Multicast Mobility Solutions in IP networks, presents a preliminary analysis of state-of-the-art and introduces the thesis' objectives.

The second chapter presents the current state-of-the-art developments in this area. This chapter also discusses multicast benefits and the current work in Mobile IP.

The third chapter presents the in-depth study of the proposed multicast mobility solution. The MTAMM's architecture is presented and explained. The several agents, as well as the protocol that supports the MTAMM are also described in this chapter.

The fourth chapter presents the Architecture implementation. Network Simulator (NS) is generically presented. Detailed information about modifications that were performed is explained. The development of agents and protocol which feature better transparency and efficiency, designed to support Mobile Terminals mobility with further efficiency and scalability, is deeply presented.

The fifth chapter presents performance results via simulation studies, of the base protocol. The evaluation of the efficiency and scalability of MTAMM is here presented.

The sixth and last chapter summarizes the main results and contributions of this thesis and provides directions for further work.

Finally, the main text is complemented with a series of appendixes that cover in more detail additional topics.

2 – Background and Related Work

Along these years, development of multimedia applications has grown significantly. These kinds of applications restrict the bandwidth occupation and present delays requirements. In order to avoid redundant packets in the network, which causes performance loss, the use of multicast connections needs to be in place. Furthermore, equipments tend to be smaller and more portable. We see more people using their self phone to watch TV, or their notebooks or even their Personal Digital Assistant (PDA) to participate in video-conferences. This way, this chapter aims to demonstrate that, since multicast routing introduces benefits with multimedia applications and taking into account that these applications are, nowadays, expected to be used in mobile equipments, it is obvious that multicast mobility solutions are welcome. The chapter will introduce the concept of multicast; the paper of Mobile IP (MIP) in mobile networks as well as how integrated is multicast in this concept. Some current solutions will be presented and discussed.

In Section 2.1 – Multicast Concept and Protocols, multicast as well as the protocols that support its functionalities will be presented. Section 2.2 – Mobile IP and Multicast Mobility, introduces mobile IP. This section will explain how MIP leads with multicast terminals movements. Also in this section some current proposed solutions that extend MIP will be analysed. Finally, Section 2.3 – Summary, resumes this section.

2.1 – Multicast Concept and Protocols

Depending on the type of application, packets could be sent over a network using unicast, broadcast or multicast connections [17]. Unicast is used when connection is between one sender and only one listener. Usually, unicast connections use TCP. TCP is, by its own nature, unicast oriented, and have as one advantage the fact that it guarantees datagram delivery between sender and listener thanks to acknowledgement messages exchanged between them. Unicast connections can also use UDP, but UDP supports a lot more paradigms than TCP, and is usually associated with broadcast and multicast connections. By another hand, a broadcast connection allows sending to all hosts in a certain network. Packets will be sent only once and every host will receive them as they are sent to the broadcast address. However, for certain applications it is not desired that all the hosts in the network receive the corresponding packets. Furthermore, broadcast connections works fine with host in the same network but it is not scalable since it does not allow sending packets to other LANs. Multicast connections are centred in the concept of group and great benefits with multimedia applications.

Multicast [18] is the delivery of information to multiple listeners without the need to send several times the same data. Thus, it is more efficient and has a lower cost since it consumes less bandwidth. With unicast connections, for each listener there is a flow of information. This way, when having many host interested in receiving the same data, the data source will send many copies, one for each interested host. This is not the best solution with multimedia applications because it will have a big influence in network performance. With multicast, the source only sends that data once. Listeners that desire the corresponding datagrams have to join the multicast group. Routers in the network just duplicate packets when necessary in order to forward them to those who asked for, those who belong to the group. Thus, it becomes more efficient because each network link only contains one copy of each packet, and that information is divided into several directions when needed.

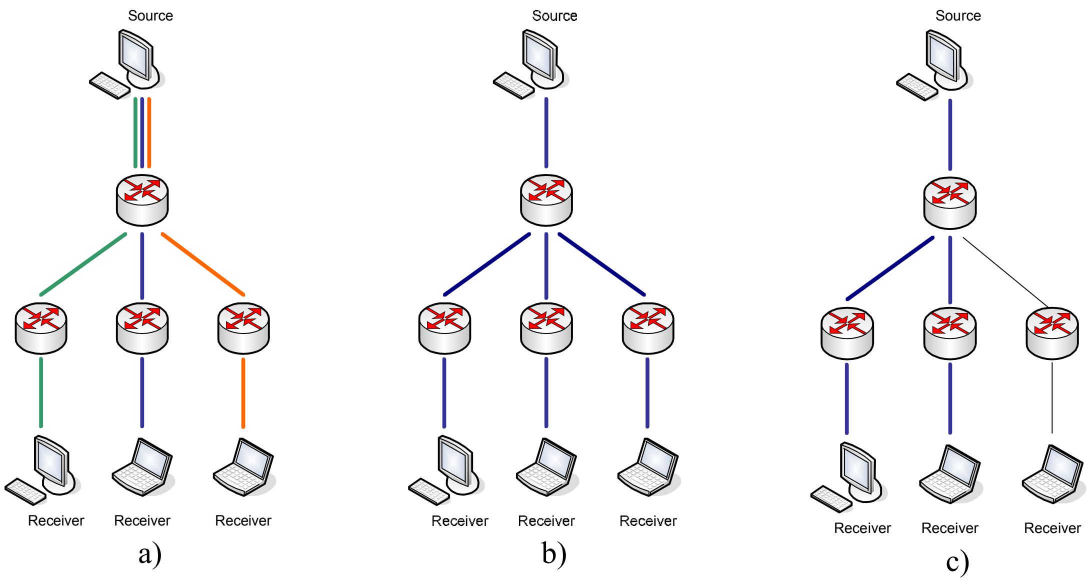

Figure 1 – Unicast (a), Broadcast (b) and Multicast (c) Connections

The whole Multicast process is complex and requires control mechanisms between network equipments. The first requirement is that routers have to be able to forward multicast packets. Multicast routing is done using protocols such as DVMRP [8] (Distance-Vector Multicast Routing Protocol), MOSPF [28] (Multicast OSPF), CBT [12] (Core Based Tree), or, the most commonly used, PIM [10] (Protocol Independent Multicast). Resuming how multicast works, using one of these protocols, a multicast tree to a certain group is build. Routers keep information about the interface they should forward packets belonging to a certain group in order to make them flow through the network in direction to listeners who join that group. This way, communication between routers is assured by protocols such as PIM while communication between routers and hosts, or vice-versa, is done using IGMP [18], in IPv4 networks, or MLD [19], in IPv6 networks. With one of these protocols, depending on IP version, sources inform their neighbour router that they will send data to a group. Listeners, use the same protocol but to inform their neighbour router that they wish to join a group. By its turn, routers use, for example, PIM to create or extend the multicast delivery tree of the desire group and forward packet to the hosts in the network. When a user aims to stop receiving the data flow it leaves the group sending an IGMP/MLD message to the neighbour router. This way, if the router does not have more listeners in its sub-network, it informs its above routers, sending a PIM prune message, to stop forwarding packets in its direction.

The source does not necessary have to belong to the group and also listeners can be members of many groups. Each group is represented by a multicast IP address from a well-defined range. In an IPv4 network, multicast IP uses the addresses of class D (224.0.0.0 to 239,255,255,255), while in IPv6, multicast addresses start with the prefix ff00:: / 8. With the introduction of IGMPv3 (IPv4) or MLDv2 (IPv6) which derives from IGMPv3, listeners can specify the source or sources from which they want to receive.

So, multicast IP is the transmission of an IP datagram to a group with one or more interested hosts (or even none), identified by a multicast IP address. Because it runs over UDP, multicast connections have no delivery guarantees, so, datagrams can reach the several listeners for the group in different orders.

2.2 – Mobile IP and Multicast Mobility

Today, multicasting introduced into the IPv4 networks, is perfectly deployed in the IPv6 fixed networks. However it constitutes a major problem for IPv6 mobile networks.

In the 1990s a network-layer 'Mobile IP' solution was first developed in the context of IP version 4 (IPv4) [29]. During this same period of time, the Internet Engineering Task Force (IETF) began work on a new version of IP that has become known as IP version 6 (IPv6) [1]. Although IPv6 addressing still retains much of IPv4's semantic link between location and identity, experience with Mobile IPv4 allowed the IETF to integrate better support for Mobile IP into IPv6. Mobility Support in IPv6 Networks [2] is expected to become a proposed standard within these days. The emerging next generation Internet infrastructure will then be ready for implementation of an elegant, transparent solution for offering mobile services to its users. However the multicast functions constitutes a particular service, unfortunately not explicitly taken into account by Mobile IPv6. The challenge of supporting voice and videoconferencing (VoIP/VCoIP) over Mobile IP remains, as current roaming procedures are too slow to evolve seamlessly, and multicast mobility waits for a convincing design beyond MIPv6.

2.2.1 – Mobile IP

Mobile IP solves the problem introduced by the fact that traditional IP addresses simultaneously represent the host's identity and encode the host's topological location on the IP network. Moving a host's physical attachment point to an IP network often results in the host moving to a new sub network with respect to the network's IP topology. When this occurs, a new IP address must be assigned to the host in order that packets may be correctly routed to the host's new location. However, since the host's IP address is also used as a transport level end-point identity such a move breaks any transport layer connections, like TCP, that were active at the time of the move. Packets being sent to the host's previous IP address are simply lost, and the host's previous peers will not know the new address to which they should now send their IP packets. Mobile IP works around this problem by introducing two IP addresses for mobile hosts – a static 'home address' by which the host is known globally, and a temporary 'Care-of Address' by which the host is known when attached to a different access router. Dynamically managed IP-in-IP tunnels (in Mobile IPv4) and specially encoded packet-forwarding rules (in Mobile IPv6) allow the mobile host to appear accessible from its home address even when actually attached to the Internet at its foreign address. Mobile IP supports network layer mobility in a manner that is transparent to all upper layers. Thus, applications designed around the assumptions of the traditional, non-mobile Internet will continue to function even in a mobile host environment. The goal is to allow applications on a mobile IP host to keep on communicating with other hosts while roaming between different IP networks. Roaming typically occurs when the mobile host physically moves to a new location and decides to utilise a different access link technology. With standard IPv4 or IPv6 such a move would result in disruption to the mobile host's communication. Using Mobile IP only a short disruption is perceived.

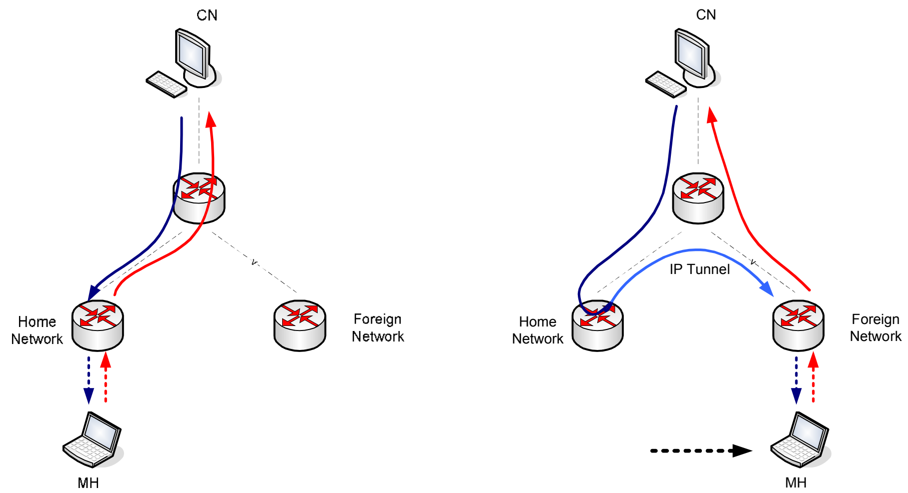

Figure 2 – Mobile IP

2.2.2 – Multicast Mobility Solutions

Because the multicast function was not explicitly taken into account by Mobile IPv6 [2], multicast faces serious problems in this type of environment. MIPv6 introduces bi-directional tunnelling (MIP-BT) as well as remote subscription (MIP-RS) as minimal standard solutions. Various publications suggest utilizing remote subscription for listener mobility only, while advising bi-directional tunnelling as the solution for source mobility. These two approaches as well as one more are common in Mobile Multicast [20].

2.2.2.1 – Bi-directional Tunnelling approach (BT)

In BT approach, the mobile node tunnels all multicast data via its home agent. When a mobile multicast source aims to redirect its multicast flow through the home network, it must tunnel the data to its Home Agent (HA). The HA receives the multicast packets from the tunnel and sends out the packets using IP Multicast Routing on behalf of the mobile multicast source. This fundamental multicast solution hides all movement since Home Agents remain fixed and results in static multicast trees. It may be employed transparently by mobile multicast listeners and sources, at the cost of significant performance degradations due to the overhead on the network and also the delay on the data delivering.

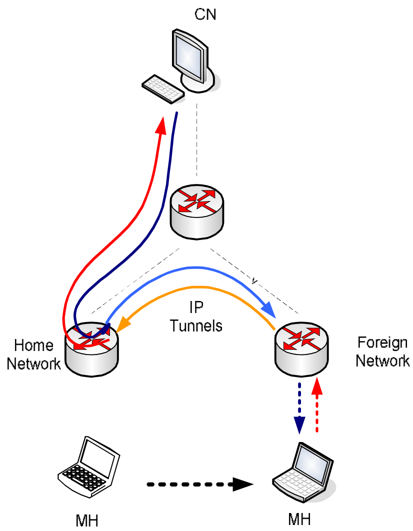

Figure 3 – Bi-directional Tunnelling approach

2.2.2.2 – Remote Subscription strategy (RS)

In RS strategy, mobile nodes always utilize their link-local addresses, thereby displaying all movements to multicast routing. Each MT re-subscribes to its desired multicast group when it enters a foreign network. Therefore a multicast router in each foreign network the MT visits must be added to the multicast tree. Multicast packets are sent directly to the local multicast router using IP multicast. The advantage of this protocol is that it provides optimum routing. However, depending on the frequency of handoffs, there may be significant overheads associated with reconstructing the delivery tree. This strategy forces the mobile node to re-initiate multicast distribution subsequent to handover. This approach of tree discontinuation relies on multicast dynamics to adapt to network changes. It not only results in rigorous service disruption, but leads to mobility-driven changes of source addresses, and thus cannot support session persistence under multicast source mobility.

2.2.2.3 – Agent-based solutions

Thereby, since these both approaches have their disadvantages, Agent-based solutions attempt to balance between the two mechanisms. Static agents typically act as local tunnelling proxies. Different types of Multicast Agent (MA) approaches have been proposed in order to reduce the reconstruction of the multicast tree. These agents join the multicast group on behalf of the multicast listeners along the different networks providing multicast source movement transparency. When a multicast source moves, and changes its current IP address to a new CoA, the multicast tree only needs to be re-established from the MA to the multicast source. Therefore, this reduces the multicast tree reconstructing and consequently the service disruption time. Unlikely, many of the proposed protocols are based on foreign agents and also on Mobile IPv4 paradigm. As the Mobile IPv6 does not have any foreign agents, these protocols cannot be directly derived to IPv6 multicast mobility scenarios. There are some alternatives or extensions to the basic Mobile IP and IP multicast interoperability approaches proposed by the IETF. Each one of these examples improves on the basic mechanisms but further refinement is needed before these solutions can be widely deployed.

Mobile Multicast (MoM) Protocol:

MoM Protocol [21, 22] is based on MIP-BT and its key extension is the use of a Designated Multicast Service Provider (DMSP) in order to solve tunnel convergence problem. A DMSP for a given multicast group is an HA chosen by the visited subnet's FA out of the many HAs that forward packets for the specific group to the visited subnet.

Mobile Multicast with Routing Optimization (MMROP):

MMROP [23] is based on MIP-RS and introduces the Mobility Agent (MA) entity to ensure routing efficiency and no packet losses from roaming. MAs are FAs that route missing packets (via tunnelling) to neighbouring subnets.

Constraint Tree Migration Scheme (CTMS):

CTMS [24] is an attempt to design a new global multicast routing protocol that would improve on Core based Trees (CBT) when it comes to highly dynamic multicast configurations, such as those found when multicast listeners are mobile. CTMS automatically migrates multicast trees to better ones, while maintaining the QoS guarantees specified by mobile users.

Multicast Scheme for Wireless Networks (MobiCast):

MobiCast [25] is based on MIP-RS and its key extension is the introduction of the Domain Foreign Agent (DFA) which serves many small adjacent wireless cells. A hierarchy is introduced, with small cells being organized into one Dynamic Virtual Macrocell (DVM). Micromobility is thus hidden from the global multicast mechanism, which does not require reconfiguring when handoffs occur within the same DVM.

2.3 – Summary

In this chapter, multicast were analysed and routing protocols were presented. We saw that multicast is centred in the concept of group which is a set of interested nodes for a certain data flow. Sources only send one copy of each packet and network replicates it only when necessary. This schema has great benefits in network performance. Then, MIP was introduced. We saw how MIP solves issues related to mobile nodes in unicast connections but, also, that a lot of work is needed when talking about multicast mobility. Some current solutions were introduced. Suggestions converge in Agent-based approaches and, it is following this line of thought, that a new approach for mobility support in multicast scenarios will be presented. This approach, named Multicast Teleport Agent approach for Multicast Mobility (MTAMM), aims to provide a novel architecture, inherent protocol signalling, named Multicast Discover Protocol (MDP), and new agents that will provide seamless multicast mobility between Access Routers on Next Generation Networks.

3 – Mobility Support for Multicast Sources and Listeners

Multicast is perfectly integrated in IP fixed networks. However, improvements in mobile scenarios are necessary since it presents some problems. One important point is the fact that multicast imposes a special focus on the source addresses. Applications commonly identify contributing streams through the source addresses, which must not change during sessions, and delivery trees in most protocols are chosen from destination to source.

In this chapter, Section3.1 – Implementation of Multicast Mobility Solution, will introduce barriers that arise to multicast mobility. Taking into account solutions that aim to overcome these issues, Section 3.2.2 – Multicast Discover Protocol, will present and deeply explain the multicast mobility solution developed in this MSc. Thesis. The main architecture will be presented as well as agents and inherent protocol. Finally, Section 3.3 – Summary, resumes the current section.

3.1 – Implementation of Multicast Mobility Solution

Multicast architectures are divided in two different approaches: Any Source Multicast (ASM) [32] and Source Specific Multicast (SSM) [6]. In the ASM approach, the multicast listener only specify the group it would like to join (*, G). Typically, in the ASM scenarios, all the multicast listeners are receiving from the Rendezvous Point (RP) through a shared tree. The multicast tree is centred on the RP and the multicast sources send their multicast content directly through a unicast connection to the RP that will be in charge to spread out their content for the specific multicast group. The major problem of this approach is that, the routing path of the shared tree is not always the best routing path between the source and the listener. In the SSM approach, the multicast delivering point is centred on each multicast source and multicast listener manifests its interests on a certain group and a specific source that it wants to receive (S, G). In this process, all the routing paths are optimized and the multicast delivery tree is not centred on a single point of failure. However, the SSM approach is not mobility-friendly since the multicast delivery tree is centred on the moving source. Thus, when the multicast source moves, its IP address changes and all the multicast listeners will stop receiving data because they are receiving from *old source* address (oS, G) and not from the *new source* address (nS, G).

Multicast routing protocols, such as PIM, already support mobility of multicast listeners since the delivery tree can converge to the current position of the terminal during its movement without intervening with the multicast reception of other listeners. However, after the handover, sessions must be re-established in order to receive the multicast data on the new position. Furthermore, such movements cause serious services disturbance that is not acceptable for real-time applications. Nevertheless, this problem can be avoided by using remote subscription mechanisms with Context Transfer Protocol (CXTP) [30] between ARs. When the multicast listener aims to move to another AR, its current multicast context is transferred before the movement to the new multicast-aware AR. Finally, after receiving the multicast context, the AR starts the remote subscription in order to prepare the multicast session on the foreign network. This is the kind of mechanisms that will be explored in this work. However, if this solves the problem associated with mobility of listeners, a solution to support the movement of sources has to be in place.

The transparency of MT movements is the major issue for the mobile multicast sources paradigm. When a source moves between two different Access Routers (ARs), listeners and multicast routers, should be able to receive the multicast data coming from the new Care Of Address (newCoA). Unluckily, on SSM scenarios the delivery trees are always centred on the multicast source IPv6 address and when it changes, the entire tree must be reconstructed to be compliant with the new one. Therefore, using the CoA directly on SSM multicast delivering process, the MT movement transparency will not be provided. After the handover, all the multicast listeners will need to re-establish the multicast session to the new source IP address (nS,G) to receive the multicast data. Unfortunately, the multicast tree reestablishment and routing convergence arrives at a time scale that is not acceptable for real-time applications.

Currently, discussions on various Agent-based approaches are ongoing. Suggestions converge in tunnelling multicast traffic through agents at fixed positions within the network. They diverge in the way agents are positioned, in the roles they attain and their interactions with multicast routing protocols. The MTAMM proposed in this work takes into account remote subscription mechanisms with Context Transfer Protocol, and also suggestions on agent-based approach to offers MTs mobility.

3.2 – A Novel Approach for Multicast Mobility Support

This section aims to present the Multicast Teleport Agents approach for Multicast Mobility (MTAMM). The solution evaluated in this thesis takes into account that mechanisms with Context Transfer Protocol between ARs promotes an efficient way to take care of HO before they effectively happens, thus avoiding huge disruption time for multimedia applications. One other important point is the fact that multicast delivery trees centred in the source leads to serious problem in mobility scenarios. When a source moves between different networks, the whole multicast tree must to be rebuild and all the listeners have to rejoin the source, but at its new location.

The following sections deeply explain the architecture behind the MTAMM. Section 3.2.1 – MTAMM Architecture, presents the architecture that supports the mobility approach. The several agents are presented in order to understand where they are place in the network and also their purpose. Section 3.2.2 – Multicast Discover Protocol (MDP) will introduce the control protocol that supports MTAMM.

3.2.1 – MTAMM Architecture

In order to implement MTAMM, different agents have to be developed. In this section, details about the design of the proposed architecture will be introduced.

The Multicast Teleport Agents approach presents a hierarchical architecture for typical operator network scenarios. This architecture is supported by four types of active agents: Mobile Terminal (MT), legacy Multicast Routers (MR), Multicast Teleport Agents (MTA) and the Multicast Source Discovery Agent (MSDA).

The MT supports the Multicast Discover Protocol (MDP) on behalf of MLDv2, in order to support multicast source subscription and multicast listeners discovering services. The MR will support legacy multicast routing protocols, such as PIM, and also needs to support MDP to handle multicast requests from MTs.

The MTA is responsible to handle the multicast traffic that is sent by a certain source in its domain and teleport this traffic to others MTAs that have listeners interested on the corresponding group. The Figure 4 – Multicast Teleport Agents architecture, shows the Multicast Teleport Agents network architecture. When a MTA has a source in its domain, this

MTA receives the corresponding data and act as a multicast source in the core network. MTAs, that have listeners interested in the group, join the multicast teleport channel in order to receive the traffic. Then, this traffic is forwarded to the next MR on its domain.

The MSDA is in charge to store and manage the multicast sources' location data base. This data base stores information about the source topology location and corresponding MTA. MTAs are connected to each other via a multicast channel that is dynamically assigned by the MSDA. This multicast channel can handle several MTAs from different administrative domains and it is used to teleport multicast data from one administrative domain to others.

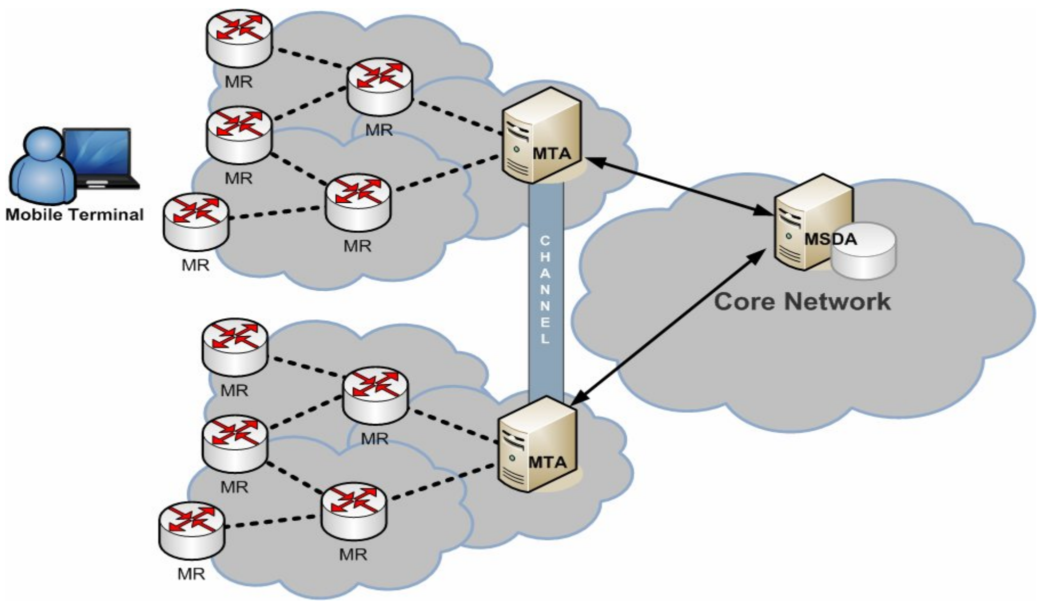

Figure 4 – Multicast Teleport Agents architecture

The MTA is a new entity designed to handle the mobility issues of multicast sources. This entity is similar to a proxy and represents the multicast source in the multicast-enabled network, with quite difference of a classical RP-based environment. The RP is the root of the multicast distribution tree and typically is a router, located on the core network and is only used in ASM multicast scenarios, or in SSM for source discovering. In the RP model, other routers do not need to know the addresses of the sources for every multicast group, all they need to know is the IP address of the RP. The MTA is an anchor point that provides a proxy features representing the multicast source on the network. The improvement is that with this solution the multicast delivery tree is not centred on only one failure point. In RP technique, the major problem is the fact that the routing path of the shared tree is not always the best routing path between the source and listeners, which leads to unsustainable delays. In the MTAMM solution, the MTA of the multicast source domain receives the multicast data from the source and teleports it to others interested MTAs through a multicast channel on the core network. On each domain that has interested multicast listeners, the correspondent MTA will be seen as the source for that stream. It will receive the teleported multicast data and will retransmit it to the multicast listeners on its domain. The multicast channel between MTAs is dynamically assigned by the MSDA during the multicast source subscription process. For each multicast source, on SSM scenarios, there is a correspondent multicast channel on the core network used to teleport data between MTAs. The usage of multicast channels inside the core network to teleport data, instead of multiple unicast flows between different MTAs, guarantees the efficient use of core network resources and simplifies data teleport process. The MTA is an anchor point which allows multicast source moving freely without changing the multicast trees of listeners. This is an important improvement, because MTAs divide the main multicast tree in two sub-trees. Since listeners are receiving from their MTA and not directly from the moving multicast source, when a source aims to move, only the multicast tree between the correspondent MTA and the multicast source must be reconstructed. With this technique the impact of moving sources in multicast scenarios is reduced increasing the efficiency usage of network resources.

Obviously, all this mechanisms require an efficient control protocol which was named Multicast Discover Protocol (MDP). This new protocol will be presented on the next section.

3.2.2 – Multicast Discover Protocol (MDP)

The MDP proposed on this multicast mobility scheme, is an access protocol that extends the functionalities of MLDv2 with mobility support. The MTs use the MDP to subscribe and discover multicast nodes on the access network.

MDP messages from the MTs point of view will provide the following functionalities:

- Multicast Source Registration: Sent by a multicast source during the start-up process. This message will make the multicast source registration on the Multicast Source Discover Agent (MSDA);
- Multicast Source Registration Acknowledge: Sent by the MSDA after succeeding the multicast source registration process and will inform the multicast source that registration was successfully done;
- Multicast Listener Request: Sent by the multicast listener to start receiving the multicast content from the MTA in its domain;
- Multicast Listener Request Acknowledge: Sent by the corresponding MTA as a response of the Multicast Listener Request message.

In order to prepare and request Handovers, the following messages are also part of the MDP:

- Multicast Source Registration: Sent by a source, under movement, to make a new registration with the indication of its new position;
- Multicast Source Registration Acknowledge: Sent by the MSDA after succeeding the multicast source re-registration process. The message informs the multicast source that the HO is prepared;
- Multicast Listener HO: When a Listener aims to move, it uses remote subscription mechanisms with Context Transfer between ARs;
- Multicast Listener HO Acknowledge: Sent by the corresponding MTA after completing the process and will inform the multicast listener that it can moves to its new location.

Multicast Discover Protocol (MDP) includes others messages used for communication between agents in the network. All the introduced process will now be detailed in order to better understand the role of each one. These processes are presented using messages flow diagrams.

When a multicast source aims to transmit multicast traffic to a group it sends a Source Registration message (*SR*) (Figure 5 – Multicast Source Registration (SR) Process) to the Multicast Access Router (Multicast AR). This message carries information about the multicast interests and eventually which Quality of Service (QoS) should be provided for the corresponding multicast session. The Multicast AR forwards the signalling message to the Multicast Source Discover Agent (MSDA) in order to register the multicast source current location. After receiving the registration message, the MSDA is able to locate the multicast source on the network and know which MTA is responsible for it. Consequently, MSDA sends an Acknowledgement message (*SR Ack*) to the Multicast AR which, in turn, forwards the message to the multicast source. Upon receiving the message, the multicast source successfully complete the registration, and is ready to start streaming.

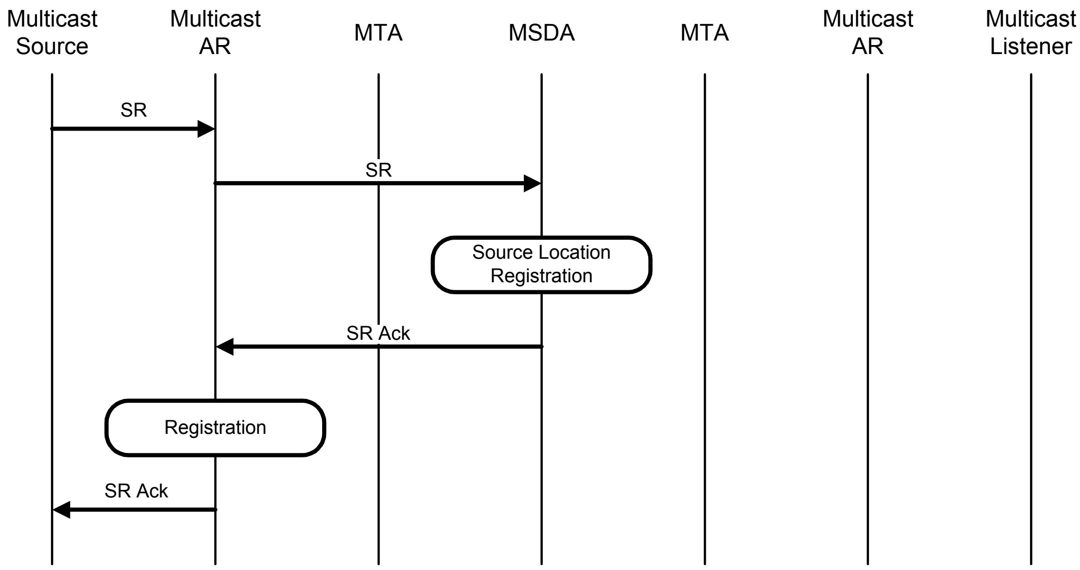

Figure 5 – Multicast Source Registration (SR) Process

When a multicast listener aims to receive from a certain group, it sends a Listener Request message (*LR*) (Figure 6 – Multicast Listener Request (LR) Process). The message is forwarded to the MTA that will use a Source Location Request message (*SLQ*) to demand to the MSDA information about the source current location. The MSDA informs the MTA with the source in its domain to start the PIM-Join process to the source in order to start receiving the multicast data. The MSDA retrieves information about the current MTA of the source via a Source Location Response message (*SLP*). Therefore, the MTA of the listener is able to join the agreed teleported channel on the core network. After starts receiving the multicast content from the teleport channel, the MTA sends a Listener Request Acknowledge message (*LR Ack*) to the AR notifying it about the success of the operation. The multicast AR starts the PIM-Join process to the MTA in order to receive the multicast data. The listener, after the reception of the *LR Ack* message, starts receiving the desired stream.

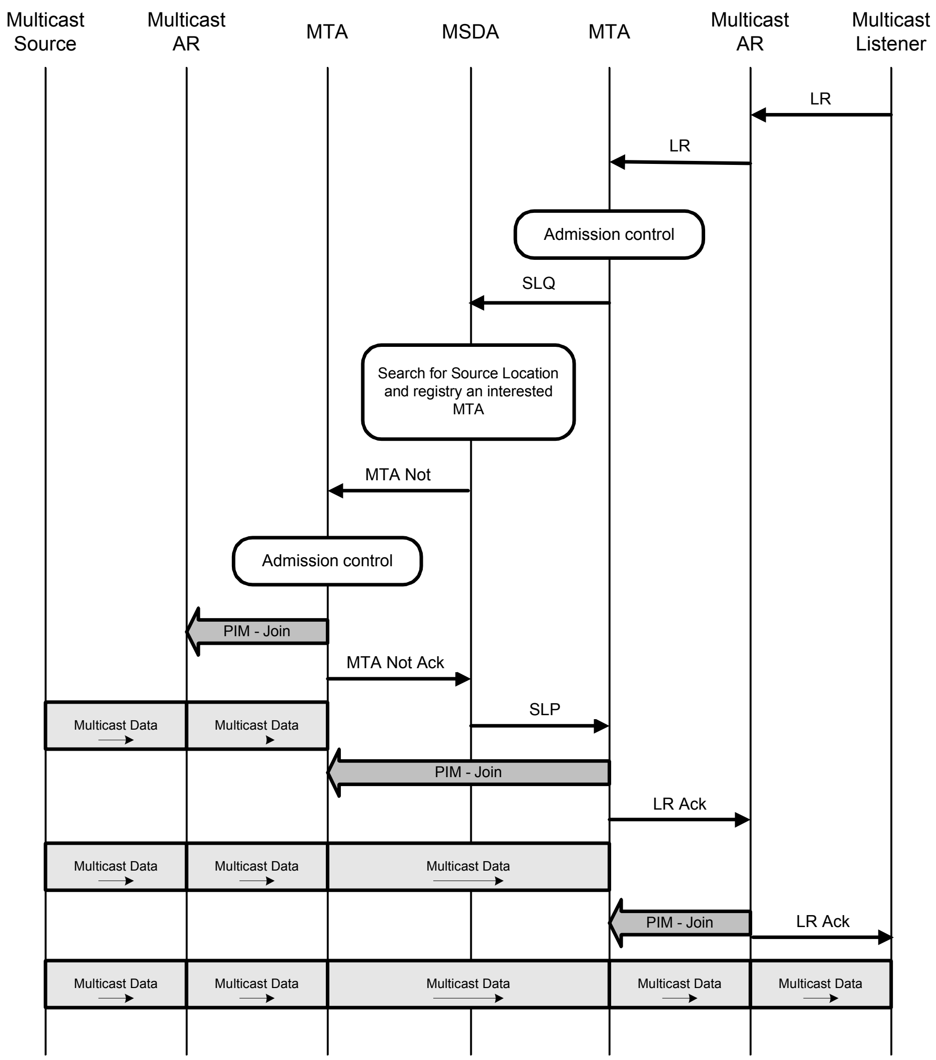

Figure 6 – Multicast Listener Request (LR) Process

The MTA architecture is designed to easily support multicast IP mobility of sources and listeners along the access network. The HO Process for both Sources and Listeners will now be presented. HO could be of two types. MTs can move inside the same MTA domain or between two different MTAs domains.

When a multicast source aims to move between two different Multicast ARs inside the same MTA domain, it performs an Intra-MTA domain handover (Figure 7 – Multicast Source Intra MTA Domain Handover Process). When the source arrives at the new AR, it sends a *SR* message. Then the AR forwards the message to the MSDA to continue the registration process. Since the registration of the source is already present on MSDA's Data Base, it refreshes the information with source's current network location and, then, sends a MTA Notification message (MTA Not) to the MTA informing it about the new location of the multicast source. The MTA starts the PIM-Join process to the multicast source’s new location. Finally, the MSDA informs the source, with a *SR Ack* message, that the HO was successfully accomplished. Once the multicast tree between the MTA and the new multicast AR will be completed, the MTA starts receiving the multicast content from the source.

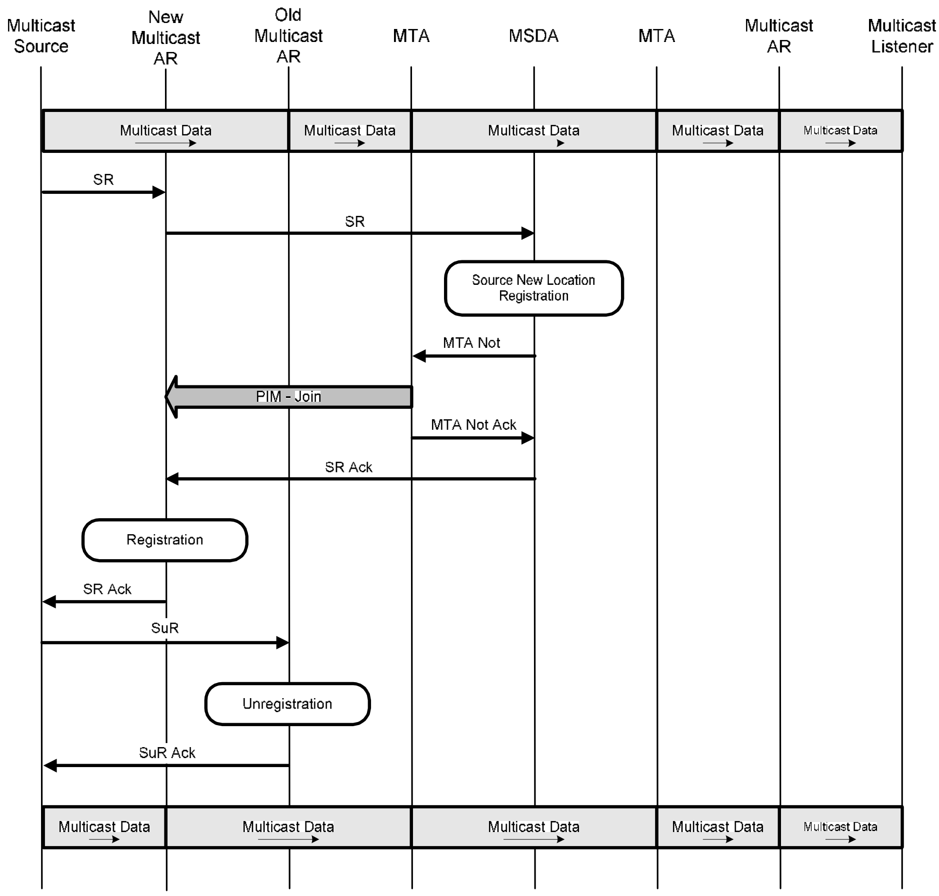

Figure 7 – Multicast Source Intra MTA Domain Handover Process

When a multicast source aims to move between two different Multicast ARs located in two different MTAs domains, it performs an Inter-MTA domain handover (Figure 8 – Multicast Source Inter MTA Domain Handover Process). The source sends a *SR* message in order to request its new registration. The Multicast AR handles the packet and forwards it to the MSDA. Since the registration of the source was already done, the MSDA knows that the source aims to realize an HO. After refreshing the new location of the source, the MSDA sends a *MTA Not* to the new MTA which, in turn, starts the PIM-Join process to the multicast source in its domain. Then, the MSDA sends a *MTA Not* to the old MTA informing to leave the source and, finally, informs the source that the HO process was successfully concluded. At the same, the MSDA advises all MTAs that have interested listeners for that multicast stream that, these MTAs, should join the new multicast teleport channel in order to receive the data coming from the new MTA. This way, listeners continue the multicast reception without the need of any operation and without changing the multicast tree.

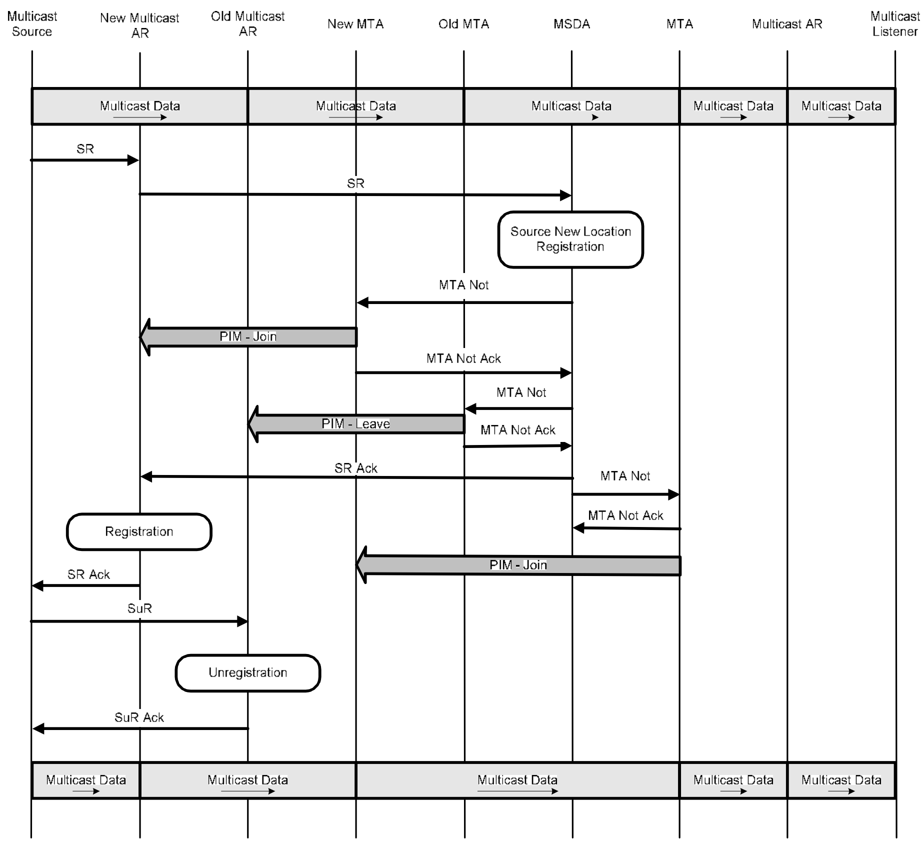

Figure 8 – Multicast Source Inter MTA Domain Handover Process

When a multicast listener aims to perform an Intra-MTA domain HO (Figure 9 – Multicast Listener Intra MTA Domain Handover Process), i.e., move between two different Multicast ARs in the same MTA domain, a *Listener Handover* message (*LHO*) is sent before start moving, to inform the AR to which it would like to move. When the current AR receives the handover request message, it realizes that the desired multicast AR belongs to the same MTA domain and forwards the message to the new listener's desired AR. The *LHO* message carries the listener's multicast context and when the new AR receives that information, it joins the multicast group according to the listener’s interest. After that process, the new AR sends a *LHO Ack* message back to the current AR. The current multicast AR forwards the *LHO Ack* message notifying the terminal that it can perform the handover. If the old AR has no more interested Listeners for that group, a PIM-Leave message is sent to stop receiving the corresponding data stream. Upon receiving the *LHO Ack* message, the HO is invoked and the listener arrives at its new AR with the multicast tree already rebuilt.

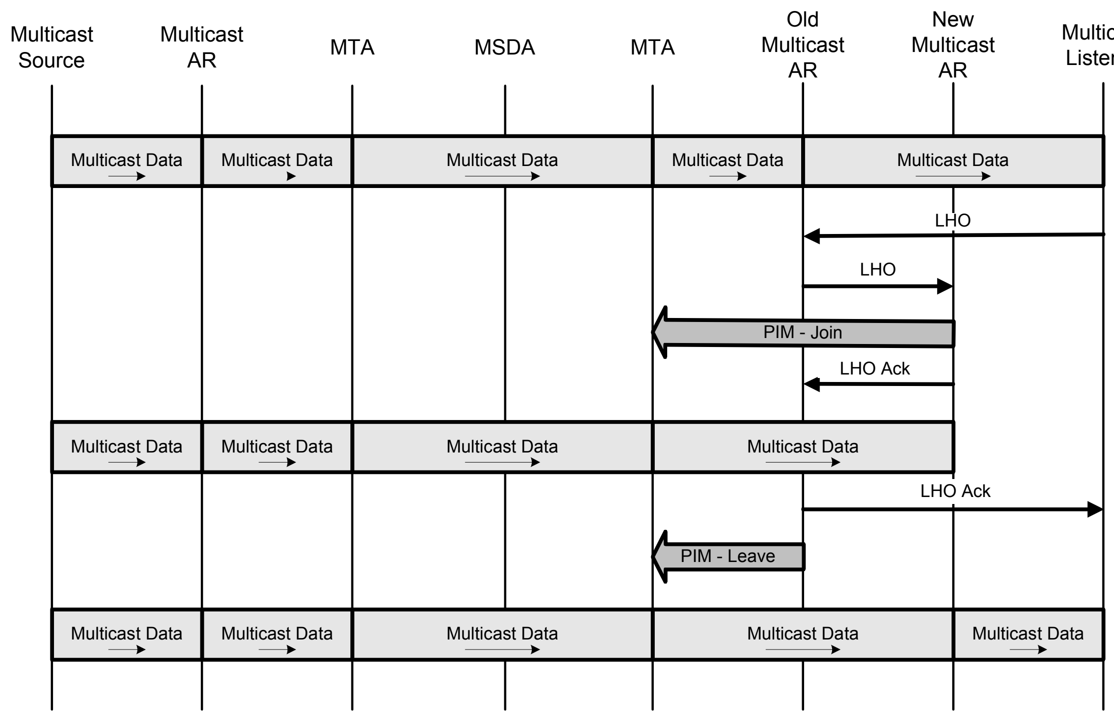

Figure 9 – Multicast Listener Intra MTA Domain Handover Process

When a listener aims to move between two different multicast ARs, a Listener HO message (*LHO*) is sent, carrying informing about the new desired AR (Figure 10 – Multicast Listener Inter MTA Domain Handover Process). When the message arrives at the current AR, it realizes than the new indicated AR does not belong to its MTA domain. This way, since the Listener aims to move between two ARs located in different MTAs domains, the listener is performing an Inter-MTA domain HO.

The current AR sends the LHO message to the corresponding MTA, requesting to lead the process. Upon receiving the message, the MTA forwards it to the MTA corresponding to the new AR. This one, in turn, after authorize a new mobile terminal in its domain, informs the MSDA about its interest in receiving the stream from the desired group and also inform MSDA that it should remove is entry about the old MTA. When the registration is completed, the new MTA joins the teleport channel on the core network to start receiving the data stream. At the same time, it forwards the LHO message to the new AR, informing that an HO will be perform and that it should join the corresponding multicast group according to the listener's context indicated in the LHO message. At the time that the listener receives the LHO Ack message, the multicast data is already received by the new multicast AR, since it already joined the group extending the corresponding multicast tree, and the disruption time is minimized.

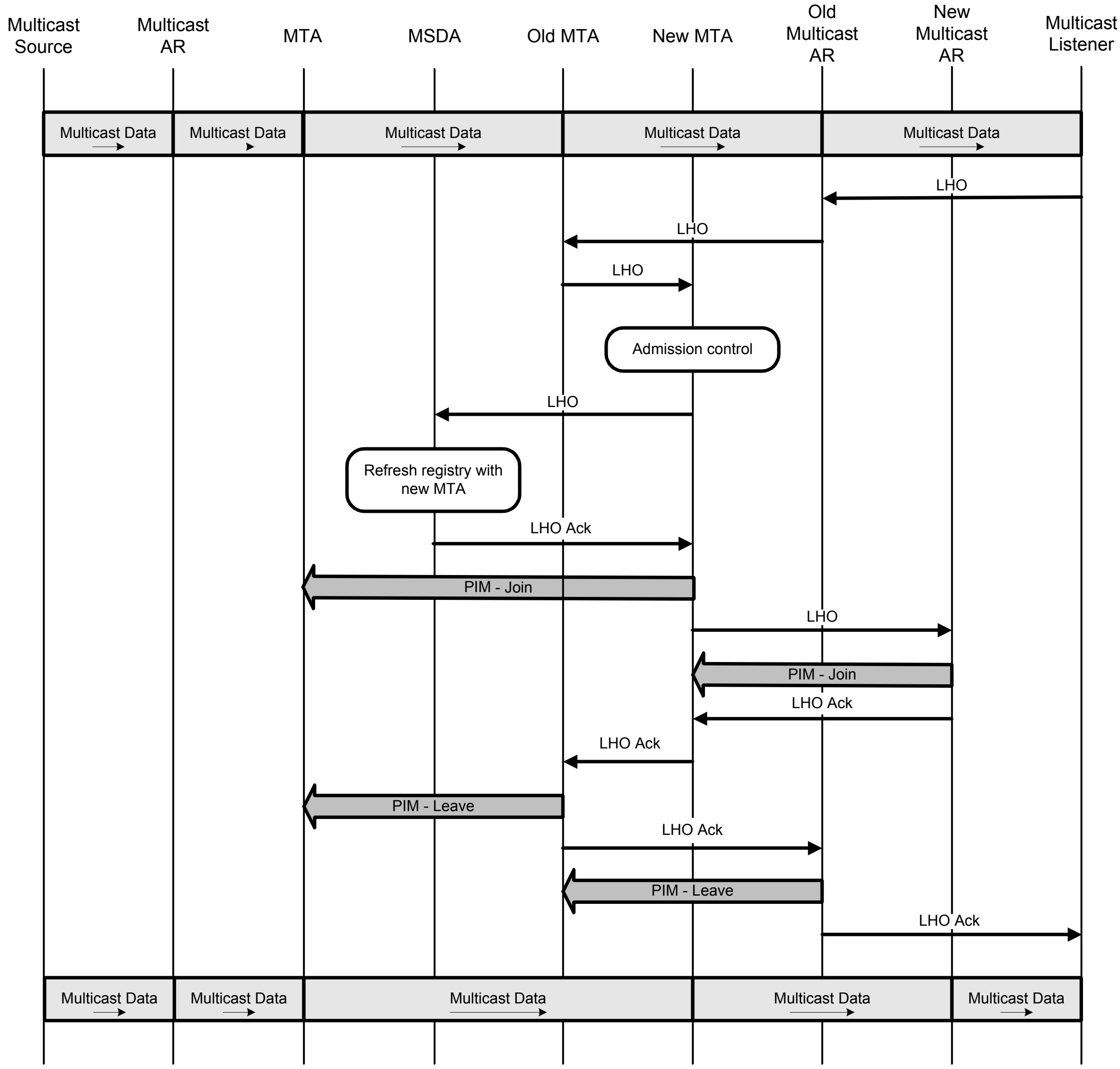

Figure 10 – Multicast Listener Inter MTA Domain Handover Process

3.3 – Summary

In this Section, the MTAMM solution was presented and analysed. Concept keys related to this approach were introduced as well as agents and protocol. The architecture includes five agents that support the MDP protocol and that have different roles in this architecture. The MSDA is an agent placed in the core network which stores, essentially, information about location of the multicast sources. This way, when a multicast source aims to transmit multicast traffic it should first perform a Source Registration Process sending a *SR* message. Listeners that are interested on a multicast stream send a *LR* message to request the corresponding data. The multicast delivery tree between sources and listeners are divided into sub-trees thanks to MTAs. The MTA is an agent responsible to act as a source or as a listener in the core network allowing the abstraction of MTs movements.

4 – Architecture Implementation

MTAMM was developed and evaluated via simulation studies. This chapter will deeply present the work that was realized and will allow us to better understand the MTAMM solution.

Section 4.1 – Network Simulator, introduces the tool that was used to develop and evaluate MTAMM. This section shows how multicast is developed in the Network Simulator (NS) and which features it offers. In Section 4.2 – First Steps, the configurations realized to simulate basic multicast and wireless scenarios are presented. This section presents why it is not possible to create and run scenarios with multicast and wireless together. After that, Section 4.3 – Extension of Simulator, explains extensions that were realized into the simulator. More properly, presents the MDP protocol that supports the MTAMM solution. Moreover, the five different agents related to this new approach will be intensively explored. Finally, Section 4.4 – Summary, resumes this chapter.

4.1 – Network Simulator

The Network Simulator (NS) was developed by UC Berkeley and allows the simulation of technologies and network protocols. NS is a discrete event simulator targeted at networking research. NS-2 was first developed in 1989 as a variant of the REAL network simulator. Currently, NS-2 has also included several contributions from researchers worldwide.

4.1.2 – Network Simulator 2.31 Overview

NS allows the evaluation and study of technologies and network protocols, as well as traffic behaviour and handling (e.g., queueing and scheduling mechanisms). The NS-2 simulator is a program developed based in object oriented languages, C++ and OTcl, and can be used to simulate unicast and multicast environments. Its code structure is divided based in processing level, which means that functions, procedures and classes that need many processing cycles must be written in C++ language. In case that the goal is to develop work that needs constant perfection and few computational processing, it will have to be developed in OTcl. Results of the simulations are written in what NS-2 calls trace files. Each line in a trace file is produced for every event of each data packet, covering all its way from the sender node to the listener node. Additionally, NS-2 can also generate trace files for Network Animator (NAM), a visual interface bundled together with NS-2 that enables users to view or record animations of a simulation.

In a simplified user version, NS-2 is a Tool Command Language (Tcl) script interpreter. It runs a simulation event scheduler, and uses network component object libraries and network setup (plumbing) module libraries. In the end, simulation results are written in trace files that can be analysed or graphically displayed. The user starts by creating an OTcl script that initiates the event scheduler, sets up the network topology using network objects (e.g. nodes and links) and creates the traffic sources that begin and end transmitting packets according to the event scheduler.

4.1.3 – Multicast in NS2

By default, nodes in NS are constructed for unicast simulations. When multicast extensions are enabled, nodes will be created with additional classifiers and replicators for multicast forwarding (Figure 11 – Internal Structures of Unicast and Multicast Nodes). Moreover, links will contain elements to assign incoming interface labels to all packets entering a node.

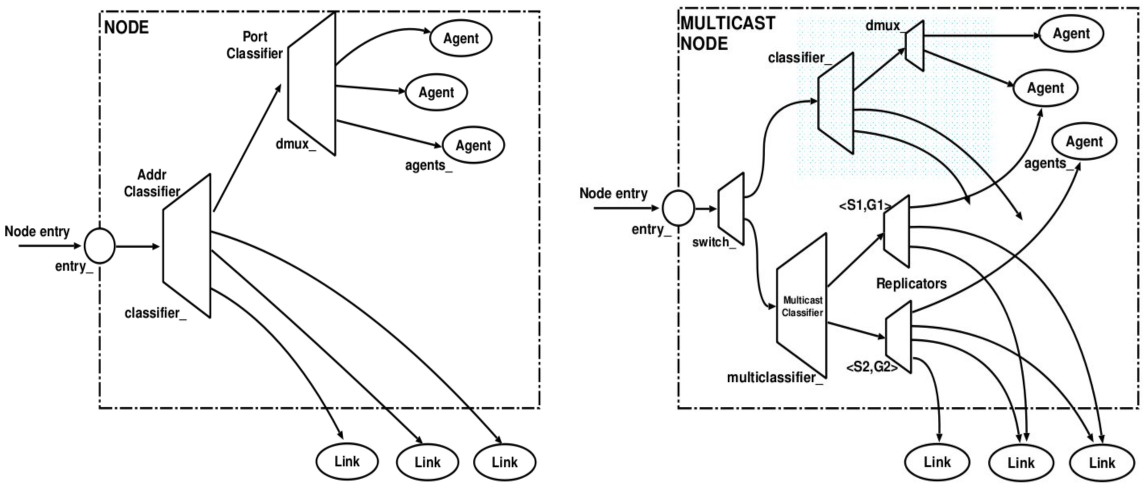

Figure 11 – Internal Structures of Unicast and Multicast Nodes

Packets are classified according to both source and destination (group) addresses. The replicator is in charge to duplicate packets in order to make them reach listeners in different locations.

Routers need a multicast routing protocol to create, extend, or leave a multicast delivery tree. NS-2 supports three multicast route computation strategies: centralised, Dense Mode (DM) or Shared Tree mode (ST).

Centralized multicast:

The centralized multicast is a sparse mode implementation of multicast similar to PIM-SM. A *Rendezvous Point* (RP) rooted shared tree is built for a multicast group. The actual sending of prune and join messages to set up state at the nodes is not simulated. A centralized computation agent is used to compute the forwarding trees and set up multicast forwarding state *(S, G)* at the relevant nodes as new listeners join a group. Data packets from the source to a group are unicast to the RP, even if there are no listeners for the group.

Dense Mode:

The Dense Mode protocol is an implementation of a dense–mode–like protocol. It can run in one of two modes. Using PIM-DM-like forwarding rules or, alternatively, can be set to dvmrp (loosely based on DVMRP). The main difference between these two modes is that DVMRP maintains parent–child relationships among nodes to reduce the number of links over which data packets are broadcasted. The implementation works on point-to-point links as well as LANs, and adapts to the network dynamics (links going up and down). Any node that receives data for a particular group without downstream listeners, send a prune message upstream. A prune message causes the upstream node to initiate prune state. The prune state prevents node from sending data for that group downstream to the node that sent the original prune message while the state is active. The time duration for which a prune state is active is configurable.

Shared tree mode:

As a version of Shared Tree Mode, NS-2 have implemented a simplified Sparse Mode multicast protocol. Class variable array RP_ indexed by group determines which node is the RP for a particular group. At the time the multicast simulation is started, the protocol will create and install *encapsulator* objects at nodes that have multicast senders, *decapsulator* objects at RPs and connect them. To join a group, a node sends a graft message towards the RP of the group. To leave a group, it sends a prune message. The protocol currently does not support dynamic changes and LANs. Bi-directional Shared Tree Mode BST.tcl is an experimental version of a bi–directional shared tree protocol.

PIM-SSM:

Since MTAMM is based on receiving the data flow from certain points on the network, more properly, the source's MTA receives the data stream from the source; MTAs which have interested listeners receives the traffic from the source's MTA; and, listeners from the MTA in their domain. This way, the multicast routing protocol to use has to allow specifying the source for the desired group. Taking into account the multicast protocols available in NS-2, apparently the solution is to use one of the protocols with RPs. However, none supports dynamic links. After some research, an implementation of PIM-SSM [27] was found which allows the specification of the source and supports network dynamic links.

4.2 – First Steps

In order to develop and evaluate the MTAMM solution some background work was necessary. This section aims to present basic configurations that were first realized with the purpose to develop wired and wireless multicast scenarios.

4.2.1 – Simple Multicast Scenario

This section describes the usage of multicast routing in NS, in particular the user interface to enable multicast routing, specify the multicast routing protocol to be used and the various methods and configuration parameters specific to the protocol.

First of all, since nodes are, by default, constructed for unicast simulations, multicast simulation should be created with the option “-multicast on”.

```
set ns [new Simulator -multicast on]
or,
set ns [new Simulator]
$ns multicast
```

Since Multicast in IP networks is based on the concept of group, to create a multicast group the following command should be used:

```
set group [Node allocaddr]
```

When a simulation uses multicast routing, the highest bit of the address indicates whether the particular address is a multicast address or a unicast address. If the bit is 0, the address represents a unicast address, otherwise a multicast address.

After that, multicast routing protocol should be configured as bellow. Scenarios were realized considering the PIM-SMM implementation mentioned in Section 4.1.3 – Multicast in NS2.

```
OSMARMcast set CacheMissMode pimdm
OSMARMcast set PruneTimeout  0.5
set mproto OSMARMcast
set mrthandle [$ns mrtproto $mproto]
```

At least, source agent and traffic applications on this agent have to be created and the destination address of the source should be the multicast address previously created. The listeners agents use the instance procedures *join-group{}* and *leave-group{}*, in the class *Node* to join and leave multicast groups. These procedures take three mandatory arguments: (*i*) the first argument is used to identify the corresponding agent; (*ii*) the second argument specifies the multicast group address; and the third argument specified the node id of the desired source.

```
$ns at 0.3 "$node1 join-group $rcvr $group [node2 id]"
$ns at 10.7 "$node1 leave-group $rcvr $group [node2 id]"
```

4.2.2 – Simple Wireless Scenario

The wireless model essentially consists of the MobileNode at the core, with additional features, that allows simulations of multi-hop ad-hoc networks, wireless LANs, etc. A MobileNode is the basic *Node* object with added functionalities of a wireless and mobile node like ability to move within a given topology, ability to receive and transmit signals to and from a wireless channel etc. A major difference between them, though, is that a MobileNode is not connected by means of Links to other nodes or mobile nodes. The four ad-hoc routing protocols that are currently supported are *Destination Sequence Distance Vector* (DSDV), *Dynamic Source Routing* (DSR), *Temporally Ordered Routing Algorithm* (TORA) and *Adhoc On-demand Distance Vector* (AODV).

The following API configures a mobile node with all the given values of adhoc-routing protocol, network stack, channel, topography, propagation model, with wired routing turned *on* or *off* (required for wired-cum-wireless scenarios) and tracing turned *on* or *off* at different levels (router, mac, agent). In case hierarchical addressing is being used, the *hier address* of the node needs to be passed as well.

```
$ns_ node-config -adhocRouting $opt(adhocRouting)
-llType $opt(ll)
-macType $opt(mac)
-ifqType $opt(ifq)
-ifqLen $opt(ifqlen)
-antType $opt(ant)
-propInstance [new $opt(prop)]
-phyType $opt(netif)
-channel [new $opt(chan)]
-topoInstance $topo
-wiredRouting OFF
-agentTrace ON
-routerTrace OFF
-macTrace OFF
```

The mobile node is designed to move in a three dimensional topology. However the mobile node is assumed to move always on a flat terrain since Z is always equal to zero. The start-position and future destinations for a mobile node may be set by using the following APIs:

$node set X_ <x1>

$node set Y_ <y1>

$node set Z_ <z1>

$ns at $time $node setdest <x2> <y2> <speed>

At $time sec, the node would start moving from its initial position of (x1,y1) towards a destination (x2,y2) at the defined speed.

In order to use the wireless model for simulations with both wired and wireless nodes, certain extensions were added to *cmu* model which is called the wired-cum-wireless feature. In order to simulate a topology of multiple wireless LANs connected through wired nodes, a BaseStationNode is created which plays the role of a gateway for the wired and wireless domains. The base station node is responsible for delivering packets into and out of the wireless domain. In order to achieve this we need Hierarchical routing. Hierarchical routing requires some additional features and mechanisms for the simulation. Therefore, the user must specify hierarchical routing requirements before creating a topology.

4.2.3 – Wired and Wireless Multicast Scenario

Section 4.1.2 – Network Simulator 2.31 Overview described that NS-2 simulator is developed based in object oriented languages, C++ and OTcl. The wireless feature introduced into NS is done using C++ language. However this thought line was not followed in multicast module development, since it is practically all written in OTcl language. This difference is responsible for a set of problems related with the *god* connection between the two languages. The information regarding the number of hops between the nodes is fed to the central object *god*. For that purpose, the following command is used:

create-god <num_nodes>

The number of mobile nodes is passed as argument which is used by *god* to create a matrix to store connectivity information of the topology.

The next figure (Figure 12 – NS Architecture) shows the architecture of the ns-allinone:

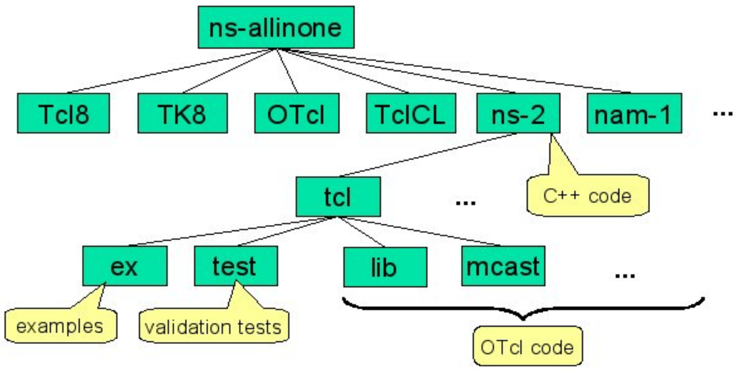

Figure 12 – NS Architecture

The figure (Figure 12 – NS Architecture) shows that Multicast in NS-2 was developed using OTcl language. Unfortunately, this leads to problems and errors while trying to configure multicast and wireless scenario together. After several days of research, no answers appear to solve this issue. The reply from the NS-2 community was clear. It's not possible unless the whole multicast code is translated to C++ language due to the problem stated before: the connection with the *god* object. This way, another solution had to be found. Since the purpose of this work is concerned to multicast mobility solutions, the only way to overcome the problem was to give up the wireless scenario. The solution followed the idea in the MIP example that comes with NS (~ns/tcl/ex/miptest.tcl). In this MIP example, a mobile node (and not a wireless mobile node) has several links, one to its Home Agent and many to its several Foreign Agents. These links are dynamic, this way, the idea is that the mobile node is connect to one of the access routers and when it aims to move the current link goes down and the link that connects it to its new access router goes up and, thanks to MIP, packet are forwarded to and from the new mobile node location. This example shows a good way to simulate nodes movement without the need of wireless. However, when talking about mobility we always associate it to wireless scenario. This way, in order to emulate wireless characteristics, in particular packets losses and delays due to collisions, ErrorModels have to be added. The ErrorModel (~ns/queue/errmodel.[cc/h]) allows some configurations that force to drop packets as well as increase the delay on a particular link. In order to make the necessary configurations, while creating the topology, the following commands sequence has to be done for each link that was supposed to be the wireless domain. Note that the rate_ variable defines the packets loss rate.

```
loss_module [new ErrorModel]
$loss_module($i) set rate_ 0.01          # loss
$loss_module($i) unit pkt
$loss_module($i) drop-target [$ns set nullAgent_]
$ns lossmodel $loss_module $n1 $n2
```

Later in this work (5.1 – Wireless Emulation), wireless simulations are studied in order to present the wireless domain behaviour when the number of MTs increases. Also in this chapter, results in wired domain, with ErrorModels, are presented.

4.3 – Extension of Simulator

This section presents the performed work and changes that were done in NS-2 in order to implement the MTAMM solution that has been deeply presented in early chapters (3 – Mobility Support for Multicast Sources and Listeners). Previously, Section 4.1.3 – Multicast in NS2 already presents the multicast routing protocol, PIM-SSM, added to NS. Section 4.3.1 – Multicast Discover Protocol, will present the MDP protocol and Section 4.3.3 – New Multicast Mobility Agents, the developed agents.

4.3.1 – Multicast Discover Protocol

In order to allow the communication between the five agents that support the MTAMM solution (see Section 3.2.1 – MTAMM Architecture for more details) some control messages have to be exchanged. The MDP (3.2.2 – Multicast Discover Protocol (MDP) features different type of messages depending on the desired process. The MDP header is presented in Table 1 – Multicast Discover Protocol (MDP) header fields.

Table 1 – Multicast Discover Protocol (MDP) header fields

msg type	Indicates the type of message depending on the desirable function
initial src addr	IP Address of the mobile host that send the message (Source / Listener)
group	IP Multicast Address
Mcast source	Multicast Source IP Address for the group (could be the Source or the MTA where it is located)
AR	The desirable AR
old AR	Used in Listeners HO to indicate the old AR
MTA	Used to indicate the MTA that transmit the multicast stream for that group
old MTA	Used in HO situations
flag g	Flag that indicates if the group is available or not
flag AC	This flag is used by MTAs to authorize or not MTs in their domain

The *msg type* field like its name says indicates the process that was performed. The initial *src addr* field is used to indicate the address of the MT that request the process. It is needed by agents such as MSDA, MTAs and multicast ARs in order to know to which MT they should send reply messages. MSDA also use this information to store in its Data Base MT's information. The *group* field indicates the multicast group address that a certain source wants to registry or that a listener aims to request. When the MSDA receives a message, like SLQ or LHO, it uses the *Mcast source* field to give information about the multicast source for the indicated multicast group. This information could be directly the source's IP address or the source's MTA IP address that is the source of the multicast teleport channel in the core network. The *AR* field is used to allow core agents to store information about the multicast AR that MTs desire. The *AR* and *old AR* fields together and also the *MTA* and *old MTA* fields are mainly used when MTs aim performing an HO. The *flag g* has many interpretations, it can be used to indicate whether a group is or not already registered in MSDA’s Data Base but is also used by MSDA, when sending MTA Not messages, to indicate if the corresponding MTA should join or leave the referred group and source. The *flag AC* is set only when the admission control of a certain MTA authorizes the MT on its domain.

4.3.3 – New Multicast Mobility Agents

The implementation of the five agents that support the MTAMM solution will be presented in the following sections. MSDA will be presented in Section 4.3.3.1, MTA in Section 4.3.3.2, Multicast AR in Section 4.4.4.4 and MTs in Section 4.4.4.5.

4.3.3.1 – Multicast Source Discovery Agent

The MSDA is located in the core network. Its purpose is to store information about Multicast Sources location. This way, when a node aims to be the source of a certain multicast flow, it sends a Source Registration (SR) message to the MSDA. In turn, MSDA stores in its database the source location, i.e., it stores information about the multicast group, source address, source access router address and the MTA address corresponding to the domain where source is located.

Table 2 – MSDA Data Base about Sources information

Group	Mutlicast Group
Source	Source IP address
AR	Source Access Router IP address
MTA	Source MTA IP Address
old_MTA	Source Old MTA IP Address (in case of Inter MTA HO)
L_MTA	List of MTAs that contains interested Listeners for that Group

Table 3 – MSDA Data Base about listeners' temporary information

Group	Mutlicast Group
Listener	IP address of the Listener that request the group
AR	Access Router IP address of the Listener that request the group
MTA	MTA IP Address of the Listener that request the group

Table 2 – MSDA Data Base about Sources information, shows the Data Base content that is stored when a source registry or re-registry (in HO situations) a multicast group. It includes information about the multicast group and its corresponding source, as well as source's network topology location such as its AR and MTA. Information about the old MTA is fielded when a source performs an HO and it is useful to achieve the HO process. The Data Base also includes information about MTAs containing listeners in their domain. This information is also used when a source performs an HO. When the MSDA informs both MTA and old MTA that a source aims to move between them, then, it informs listener's MTAs in order to make them join the new multicast teleport channel.

Table 3 – MSDA Data Base about listeners' temporary information, illustrates information that is stores when a listener aims to receive from a group that do not already have a source. When this situation occurs, the information about listeners' location as well as the requested group is stored. This information is used by the MSDA, when a source makes a registration for the desired multicast group, to advise the MTAs where interested listeners are that they can at this time start receiving the multicast data content.

The MSDA agent plays an important role in the mobility multicast Mts' control along the network. The next flow (Figure 13 – MSDA block diagram) aims to describe MSDA functions depending on the type of messages and when they are received.

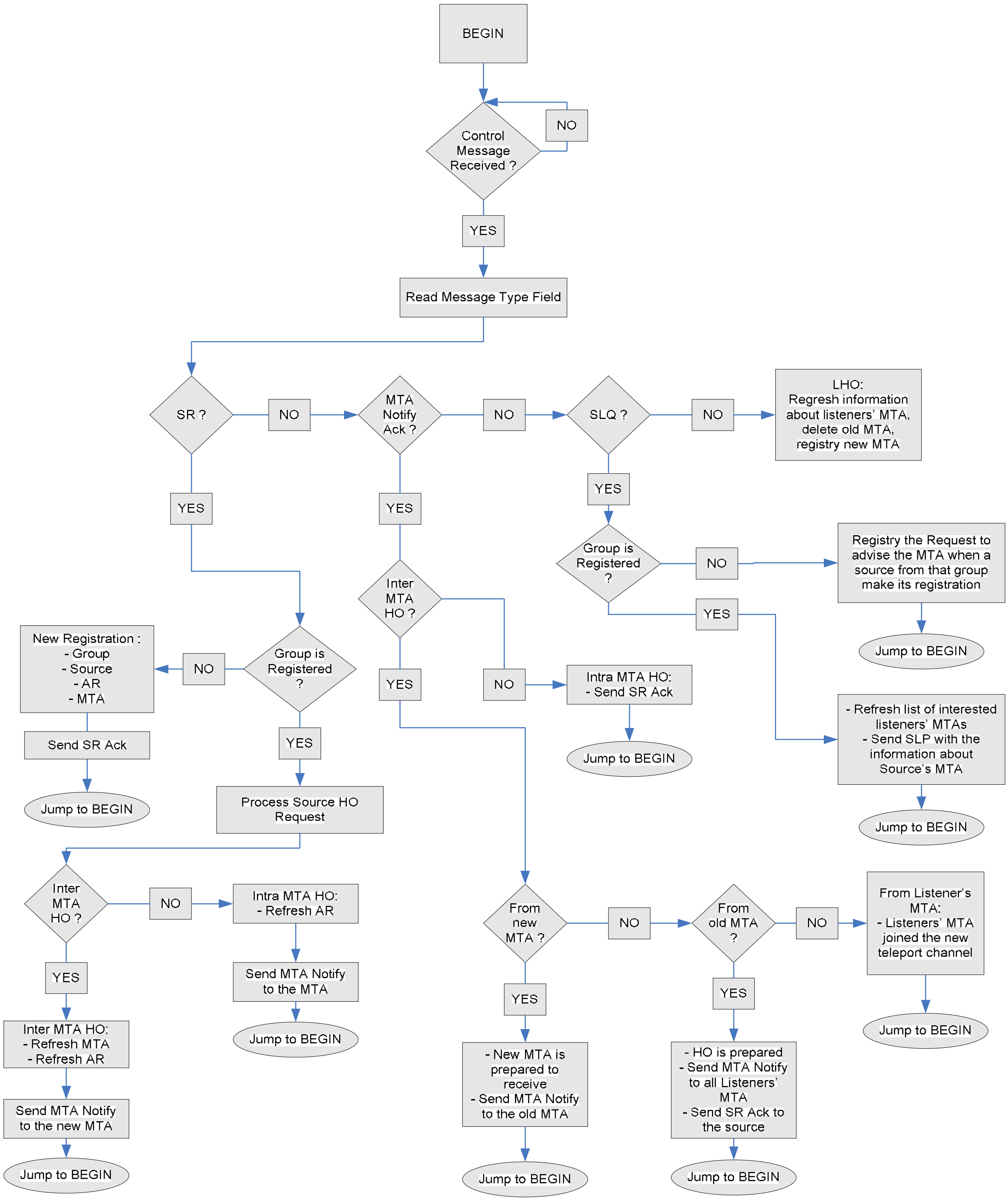

Figure 13 – MSDA block diagram

After receiving a message, the MSDA analyses the *msg type* field. If it is a SR message, MSDA will search in its Data Base if the indicated group was or not already registered. If the group does not exist, the MSDA performs the registration operation and sends a SR Ack message to the source indicating that the process was successfully done. However, if the group is already in the Data Base of the MSDA, then it realizes that the source moved to a different AR. Depending on the new AR, MSDA starts the HO process.

When the MSDA receives a SLQ message, it searches in the Data Base for the source of the desired group and replies with a SLP message. MSDA stores in its Data Base the MTA location in order to advise it if, in the future, the source changes its location.

As a consequence of Listeners' Inter MTA HO, the new MTA sends a LHO message to the MSDA. Thus, the MSDA should remove the old MTA from its Data Base and replace it with the new one.

4.3.3.2 – Multicast Teleport Agent (MTA)

The MTA is an entity essentially designed to handle the mobility issues of multicast sources. The MTA divides the main multicast tree in two sub-trees and, in fact, acts as a multicast listener and also as a multicast source. From the point of view of the original multicast source, the MTA is a common multicast listener because it joins the group and receives the multicast traffic from the source. The data content is sent to others MTAs through a multicast teleport channel in the core network and, from the point of view of the multicast listeners, the MTA is a common multicast source. In order to manage sources and listeners in its domain, each MTA needs to store information, according to Table 4 – Stored information in MTA:

Table 4 – Stored information in MTA

group	Multicast Group
source	Indicate if the Source is or is not in this MTA (Boolean)
source_addr	Source IP address if source is in this MTA; Or, MTA IP address where the Source is in order to join the teleport channel
listeners_num	Number of listeners for the group in this MTA

For each group which corresponding data is travelling trough the MTA, it keeps information about the multicast IP address of that group as well as a flag that indicates if the corresponding multicast source is or not on its domain and, if it is, registry its IP address. Moreover, the MTA stores the number of multicast listeners in its domain allowing MTAs to know when to leave the multicast teleport channel in the core network. The MTA block flow is presented in Figure 14 – MTA block diagram.

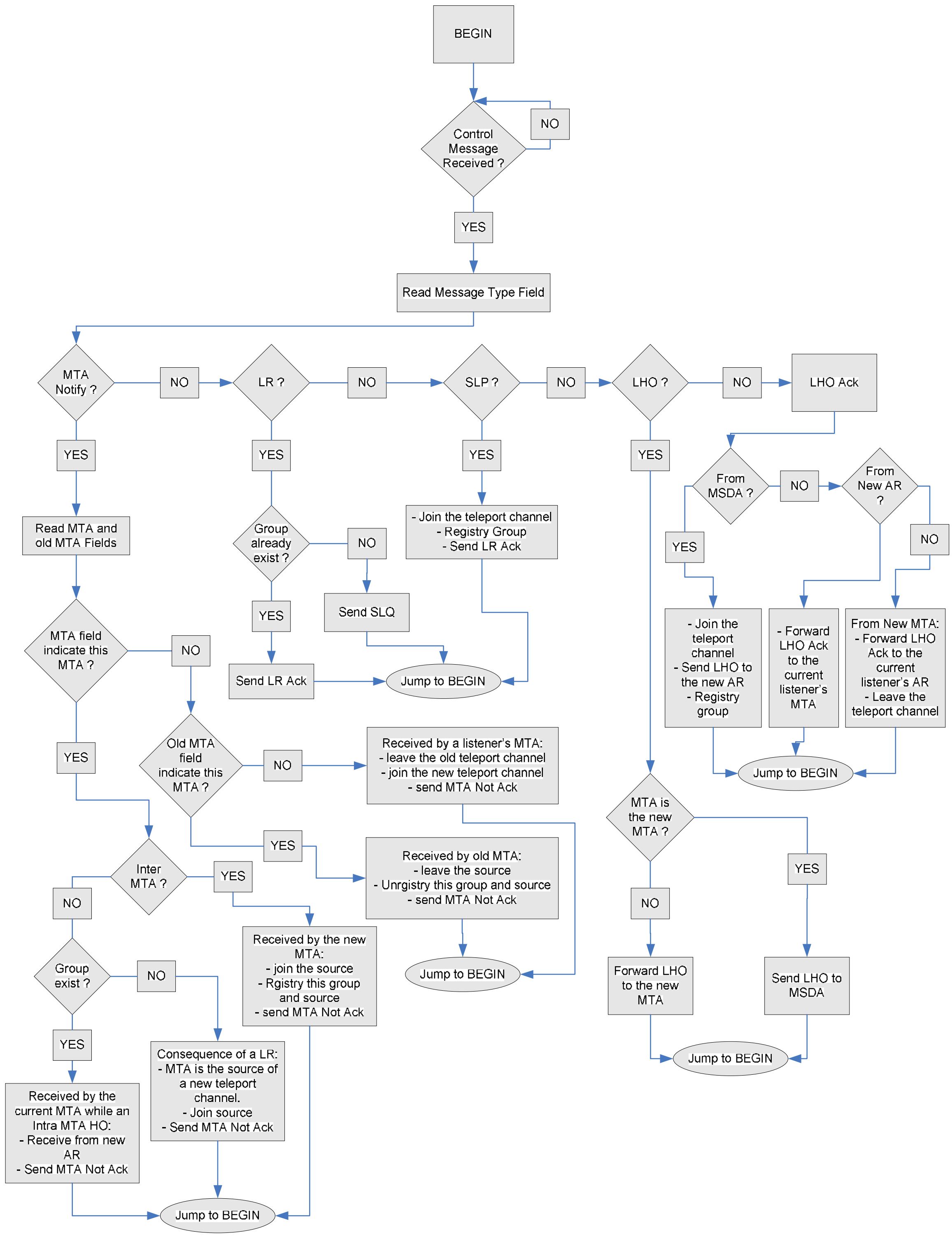

Figure 14 – MTA block diagram

Upon receiving a message, MTA reads the *msg type* field. Messages could be of five types. If it is a MTA Not message, MTA reads both *MTA* and *old MTA* fields in order to decide what operation it should realize.

When a MTA receives a LR messages, it consults its Data Base in order to verify if the indicated group is or not already registered.

An SLP message is received by the MTA, as an answer of a SLQ message, with information about source's location.

LHO messages are received when listeners decide to move, and MTAs act differently depending if they are the current or the new listener's MTA.

4.3.3.3 – Access Routers

The Multicast Access Router is the element that acts as an intermediate between MTs and the network. It supports legacy multicast routing protocols such as PIM as well as MDP in order to handle the multicast requests from the MTs. A block diagram is presented in Figure 15 – AR block diagram.

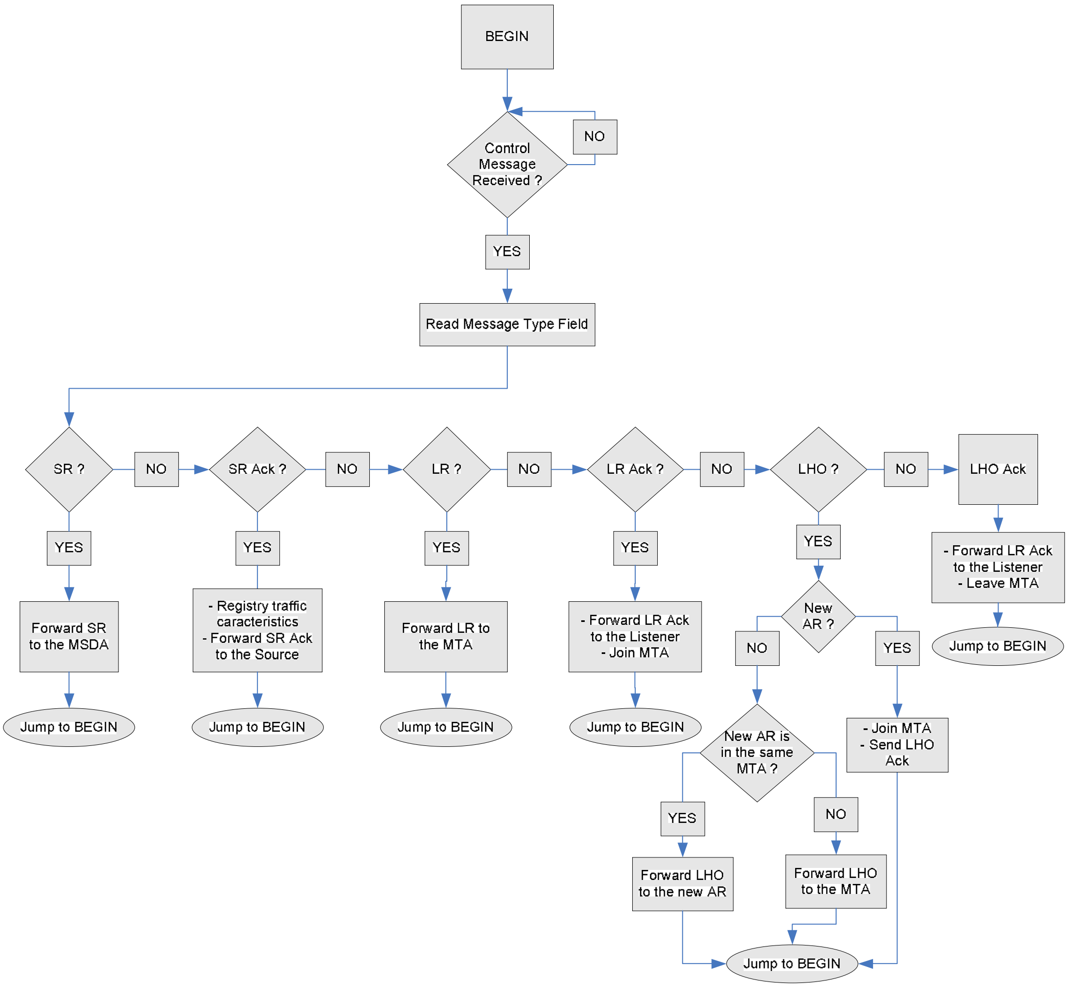

Figure 15 – AR block diagram

The AR is essentially reactive. Most of times, when it receives a message it forwards it.

When an AR receives a SR message it forwards it to the MSDA, then, when it receives the SR Ack it stores information about traffic characteristics and delivers the message to the corresponding source.

After received a LR message, the AR forwards it to the MTA in its domain. Then, when it receives the LR Ack, it starts the PIM-Join process in order to receive the data stream from its MTA, and then forwards the LR Ack message to the listener.

When a LHO message is received, the AR reads the IP address in order to take further decisions.

4.3.3.4 – Mobile Terminals: Sources and Listeners

MTs are the main active part in this whole process. They use the MDP to interact with the network. In order to perform the HO process, both sources and listeners store information about their network location, in particular, information about AR and, when they aim to move, about the old AR (Table 5 – Information stored by Sources and Listeners).

Listeners, besides this information, also keep the multicast groups as well as the IP address of the MTA that is sending the corresponding data stream (Table 6 – More information stored by Listeners).

Table 5 – Information stored by Sources and Listeners

AR	The current AR to which mobile node is connected; Or, in an HO situation it is the new AR
old AR	In HO situations it is the current/old AR

Table 6 – More information stored by Listeners

group	List of multicast group that Listeners are receiving from
Src addr	IP Address of the MTA from which Listener is receiving the traffic

Next, in Figure 16 – MTs block diagram, a block diagram will be presented with the purpose to explain sources and listeners behaviour in the MTAMM solution. The block diagram presents possible actions that MTs can make when they aim to as well as the way they react as consequence of received messages.

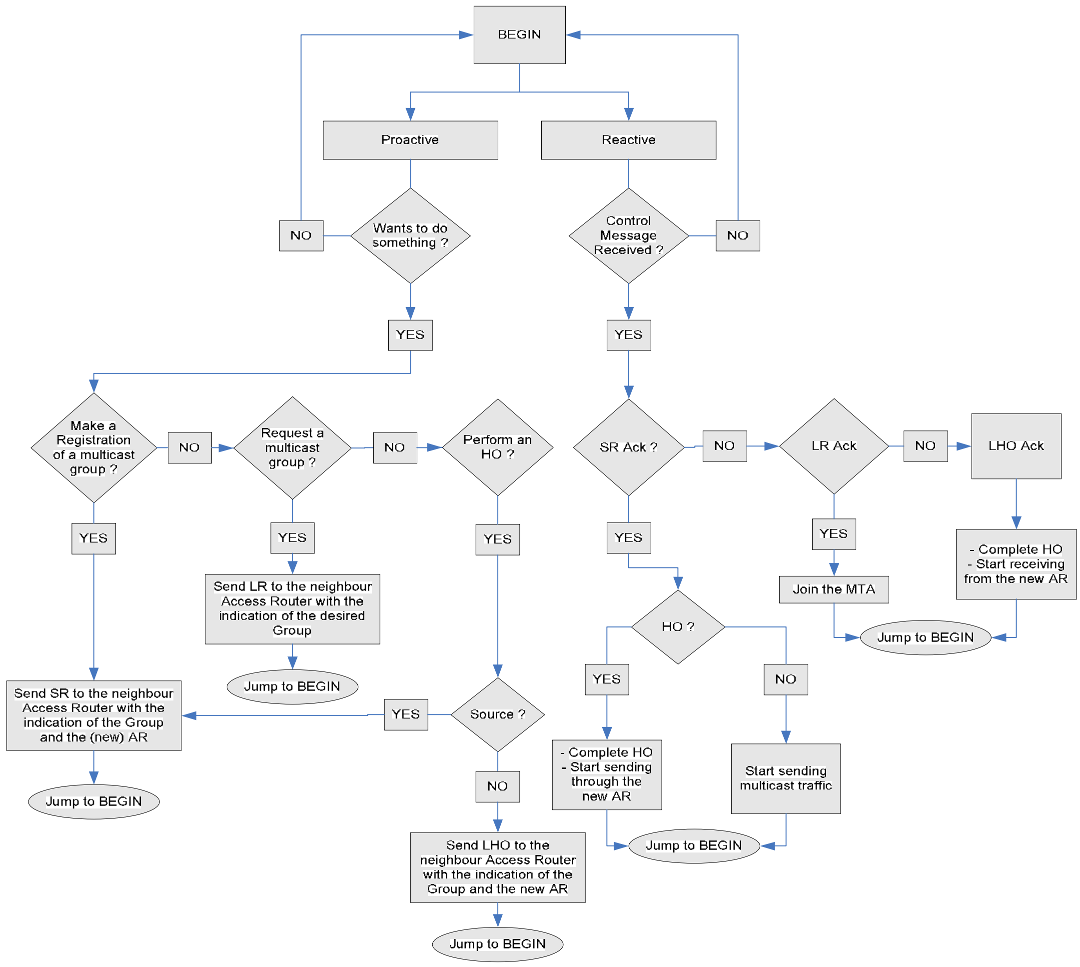

Figure 16 – MTs block diagram

A MT can perform three different actions. When a MT is a source of a certain multicast group, it sends a SR message to the network in order to request its registration. However, if the MT is a listener, it sends a LR message with the purpose to request a certain multicast group and starts receiving the corresponding data.

Both sources and listeners can move between different ARs. If the MT is a source, then it should start a new registration process by sending a new SR message with the indication of its new location. However, when it is a listener that aims to move, it sends a LHO message to its current AR in order to prepare the HO and then move to the new desired location.

4.4 – Summary

Chapter 4 presents the extensions and work done in the network simulator in order to develop and evaluate MTAMM. NS was presented since it is the tool used to the development and evaluation of the proposed approach. Then, the multicast development in NS was presented, more properly, which protocols it offers and why none of them offers exactly what was needed to develop the MTAMM solution. Section 4.3 – Extension of Simulator, deeply analyses the developed MDP protocol that supports this multicast mobility solution. Moreover, the several agents, MSDA, MTAs, ARs, Sources and Listeners were intensively explored through block diagrams.

5 – Architecture Evaluation

This chapter presents the evaluation of the MTAMM solution that has been developed and presented in this thesis.

Section 5.1 – Wireless Emulation, presents tests realized in a wireless scenario considering a simple network with four Access Routers (ARs) and some mobile terminals that will act as senders to a Correspondent Node (CN) in the core network. The purpose of this test is to extract some results and conclusions that will allow the emulation of the wireless domain behaviour. After this, a wired scenario, considering the same characteristics, will then be evaluated demonstrating that it is possible to efficiently emulate the wireless domain. Also considering the wireless scenario, in Section 5.2 – Wireless HO Evaluation, MTs were configured to act as sources or listeners to and from the CN (in unicast). This test aims to measure the latency associated to MTs movements and, in the future, in Section 5.4 – Summary, will allows the comparison between the Handover (HO) latency associated to the MTAMM solution. The evaluation of MTAMM will be performed in Section 5.3 – Evaluation of the MTAMM Solution. In these tests, a network with fifteen ARs fairly distributed along the six MTAs' domains will be considered. Taking into account different conditions and configurations, the solutions will be tested and the network performance will be analysed. Finally, Section 5.4 – Summary, summarizes this current chapter.

The extracted results allow concluding that the MTAMM solution efficiently controls MTs movements and that associated HOs are acceptable for real-time applications.

5.1 – Wireless Emulation

As we previously described (Section 4.2.3 – Wired and Wireless Multicast Scenario), currently, NS presents some incompatibilities between wireless and multicast. To solve this issue dynamic links between MTs and their ARs were used emulating a wireless network. However, additional configurations are necessary to effectively emulate the wireless behaviour. To accomplish this objective, ErrorModels have been configured in these wired dynamic links. Thus, it is possible to force packets loss and delays as exactly happen in wireless scenarios. Therefore, tests realized in a wireless network in order to extract some results and conclusions are presented in the next section (Section 5.1 – Wireless Emulation) and, furthermore, we will see the configurations that are necessary in ErrorModels and dynamic links (Table 8 – Configuration of ErrorModels) to effectively use wired links as if it was in fact a wireless network.

Consider the two following figures (Figure 17 – Wireless scenario with MIP and NOAH, and Figure 18 – Wired scenario).

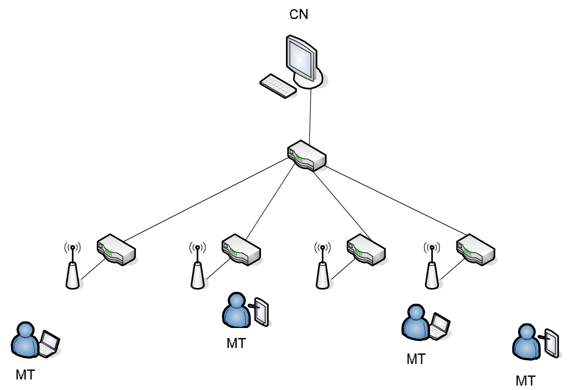

Figure 17 – Wireless scenario with MIP and NOAH

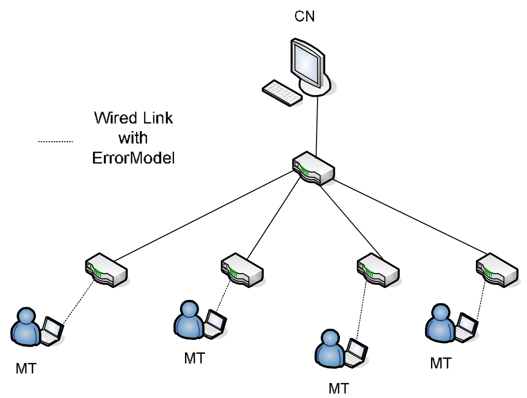

Figure 18 – Wired scenario

The figure on the left (Figure 17 – Wireless scenario with MIP and NOAH) shows the tested wireless network. This scenario was performed using the patch IST_CIMS [26] as well as Mobile IP. This patch [26] does not include the ad-hoc routing protocol, Destination Sequence Distance Vector (DSDV) for the wireless infrastructure routing protocol. This way, it requires less processing at each node, making the Handover process faster. The other figure (Figure 18 – Wired scenario) shows the same scenario, but the wireless domain was replaced by wired dynamic links with *ErrorModel*. Both scenarios include some MTs which act as unicast senders to a CN in the core network. In Table 7 – Scenarios' Characteristics, simulations characteristics are presented. Each MT has configured one Constant Bit Rate (CBR) Application that runs over UDP, which send packets with 1000 Bytes at 512kbps.

Table 7 – Scenarios' Characteristics

MTs (Sources)	ARs/APs	CBR Rate	CBR packet size	Receiver (CN)	Core Links Bandwidth	Core Links Delay
1 – 15	4	512 kbps	1000 Bytes	1	10 Mb	2 ms

The wireless scenario was performed considering a topology with 1000 x 670 meters (X x Y), and the four Access Points (APs) were fairly distributed minimizing collisions effects between them. MTs are fairly distributed along the topology and then move to a randomly selected position.

Results extracted from the wireless test will now be analysed. The number of MTs was increased from one to fifteen (1 – 15) and, consequently, the network performance is affected. The presented average values are result of five simulations for each situation with the confidence values of 90%.

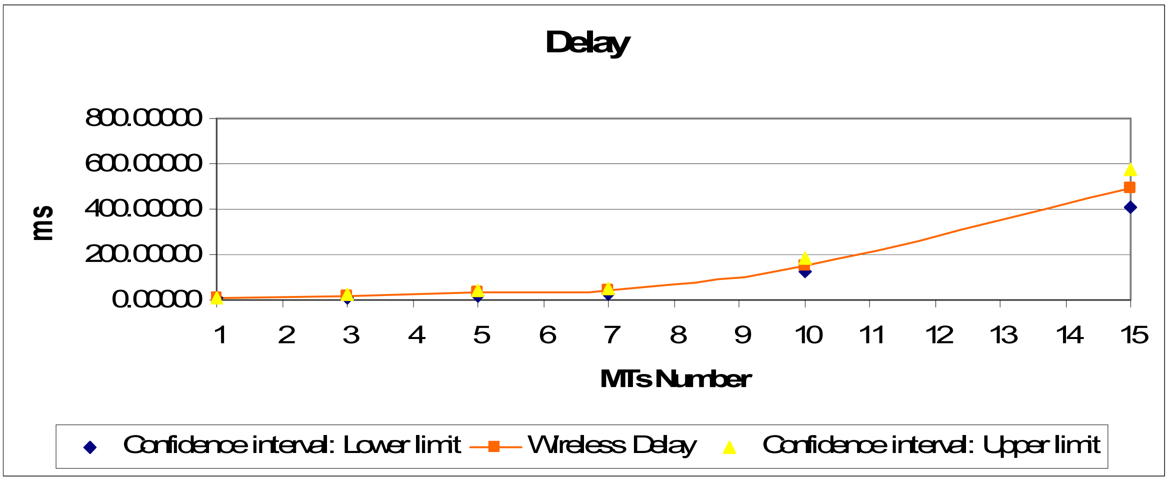

Figure 19 – Delay in Wireless scenario

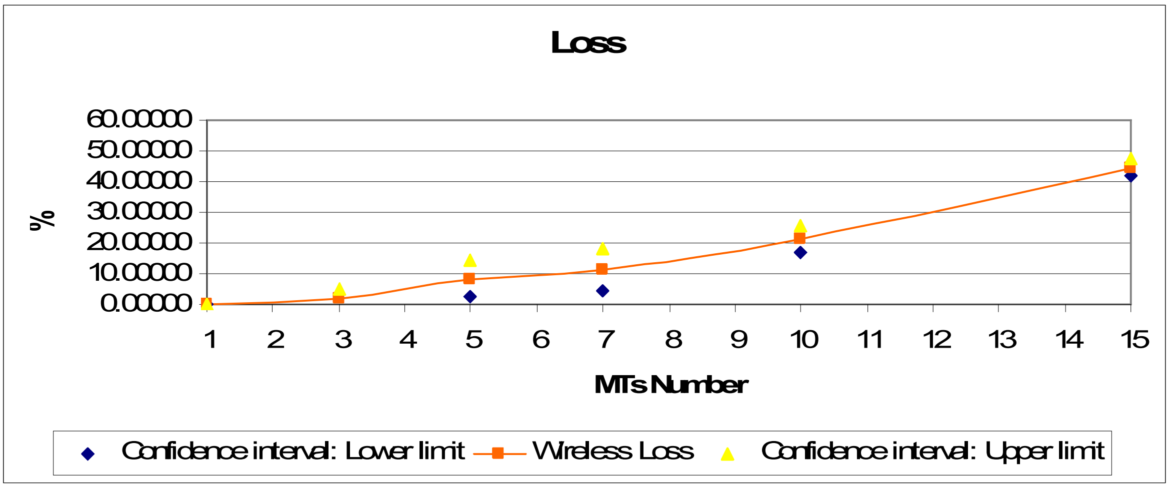

Figure 20 – Percentage of Packets Loss in Wireless scenario

The network's delay (Figure 19 – Delay in Wireless scenario) is very affected by the number of MTs. With only one MT the registered average delay was 4.6 ms, while, with fifteen MTs this value reaches 491.6 ms. This way, the difference is very significant. Such behaviour is a consequence of wireless phenomena, such as collisions. The network layer provides mechanisms to avoid these collisions, for example, messages such as Request to Send (RTS) and Clear to Send (CTS) are exchanged. However, these collisions are inevitable and imply the loss of some packets. When such situation occurs, sometimes the MT sends again the same packet increasing the delivery time. When packets are not resent, the percentage of packets loss also increases with more MTs (Figure 20 – Percentage of Packets Loss in Wireless scenario). With one MT, the percentage of collisions and loss was zero but, with more MTs, more collisions occur and more packets are loss in the wireless domain. With fifteen MTs, this value is around forty six percent (46%), which is significant.

In order to perform a fairly comparison, the same scenario maintaining the same configurations was considered while testing the behaviour of the wired dynamic network. The wireless domain was replaced by dynamic wired links with ErrorModels. It was necessary to tune each ErrorModel to offer a very similar behaviour to wireless domain. So, in order to emulate the wireless behaviour and as result of some attempts, values presented in Table 8 – Configuration of ErrorModels, were configured.

Table 8 – Configuration of ErrorModels

MTs	Link Delay (ms)	Rate Loss
1	1.0	0.000
3	10.3	0.020
5	25.0	0.075
7	35.0	0.100
10	148.0	0.175
15	486.0	0.315

Considering the network presented in Figure 18 – Wired scenario, and taking into account values configured in Error Models (Table 8 – Configuration of ErrorModels), Figure 21 - Delay in Wired scenario, aims to illustrate the wired network behaviour, emulating the wireless one.

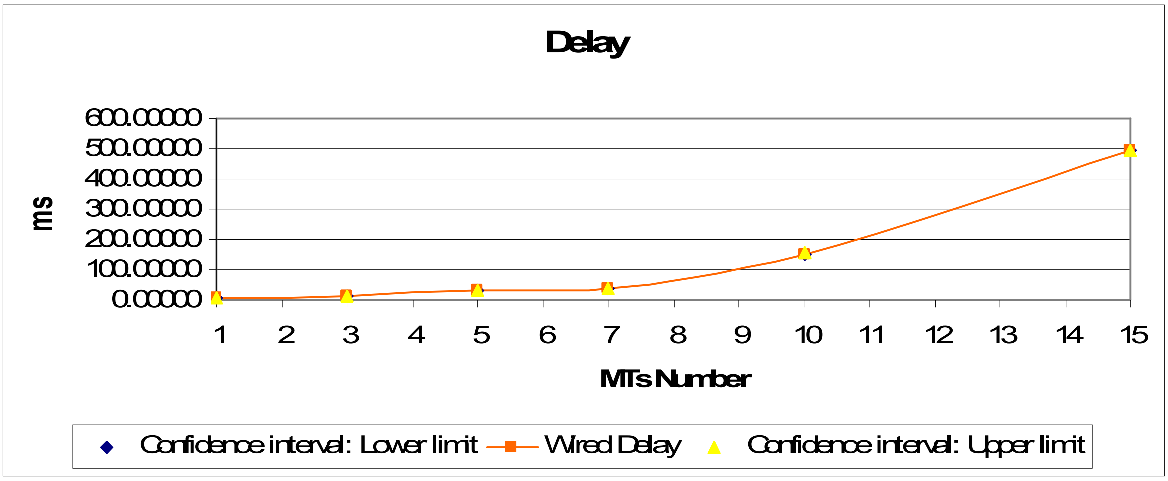

Figure 21 - Delay in Wired scenario

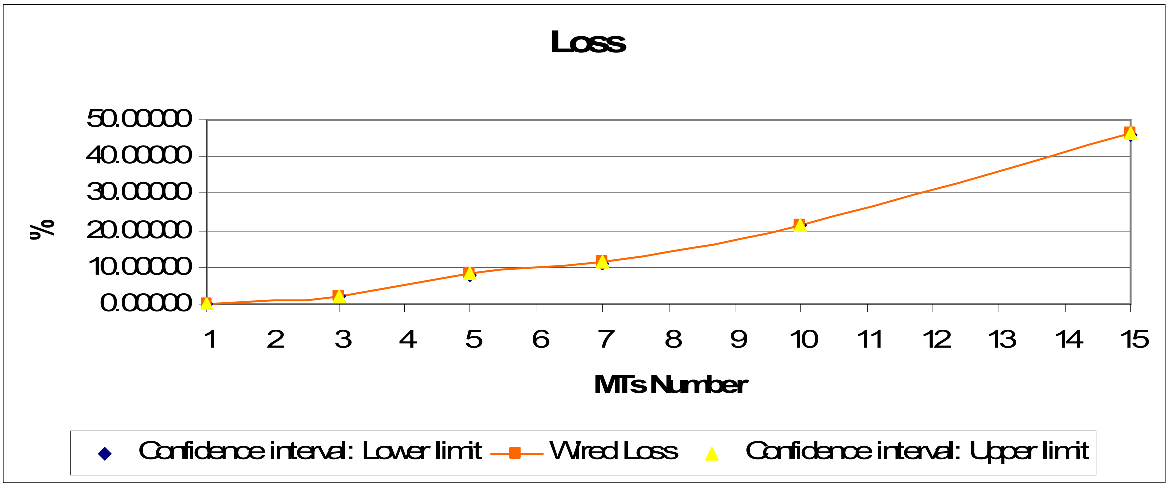

Figure 22 - Percentage of Packets Loss in Wired scenario

Analysing Figure 21 - Delay in Wired scenario, and Figure 22 - Percentage of Packets Loss in Wired scenario, the behaviour and performance of wired scenarios emulating wireless are exactly the same as pure wireless scenarios. As in wireless network, delay and percentage of Packets Loss are affected by the number of MTs.

Figure 23 – Delay Comparison between Wireless and Wired Scenarios, and Figure 24 – Packets Loss Comparison between Wireless and Wired Scenarios, demonstrate how close values are in both configurations.

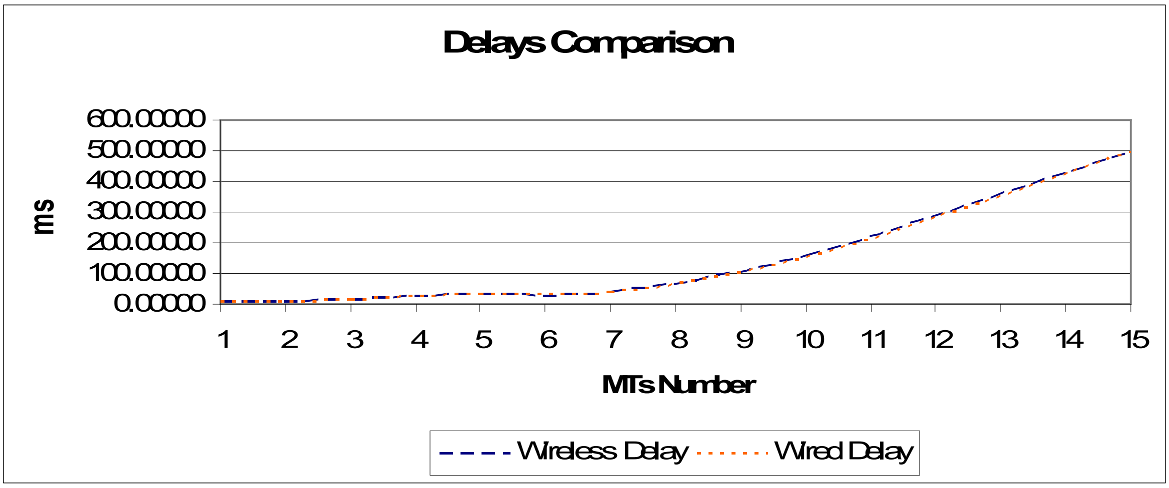

Figure 23 – Delay Comparison between Wireless and Wired Scenarios

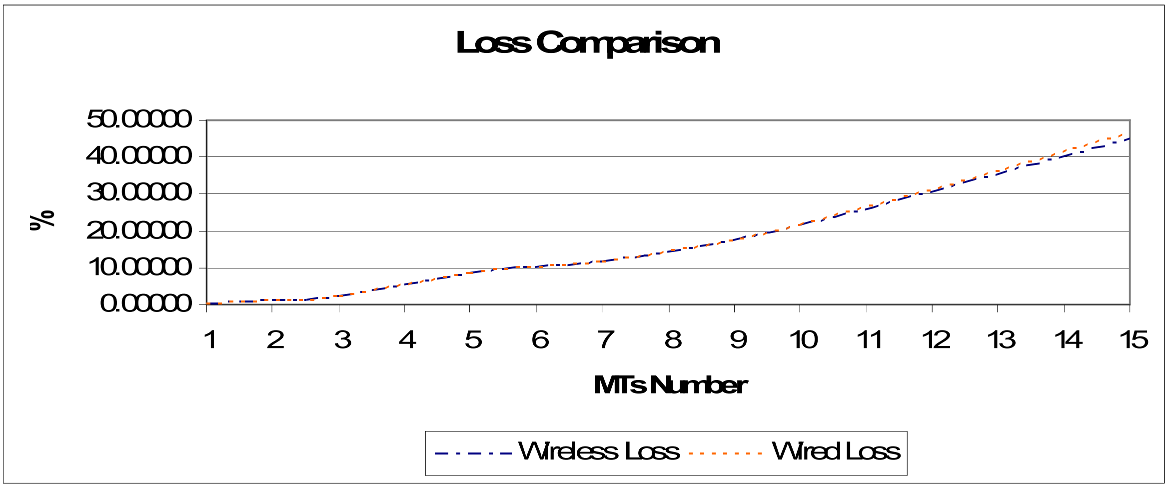

Figure 24 – Packets Loss Comparison between Wireless and Wired Scenarios

This way, the introduction of *ErrorModels* in wired links really gives the possibility to simulate MTAMM using dynamic links exactly as if we were using a wireless topology.

5.2 – Wireless HO Evaluation

As was previously described in Section 5.1 – Wireless Emulation, sources are fairly distributed around the topology. When each source starts moving, packets will be forwarded through the new AP and, as consequence of these movements and collisions between the several MTs on the network, some packets are loss. In Figure 25 – Sources' HO Latency in Unicast Connections with MIP and NOAH, the time values presented correspond to the HO latency in wireless scenario, i.e., the time interval between the last received packet coming from the old AP and the first received packet, after the Handover process has been completed, coming from the new AP. The purpose of these results is to compare values of the HO latency in this unicast wireless scenario with those extracted from the MTAMM solution that will be presented in the next section (Section 5.3 – Evaluation of the MTAMM Solution).

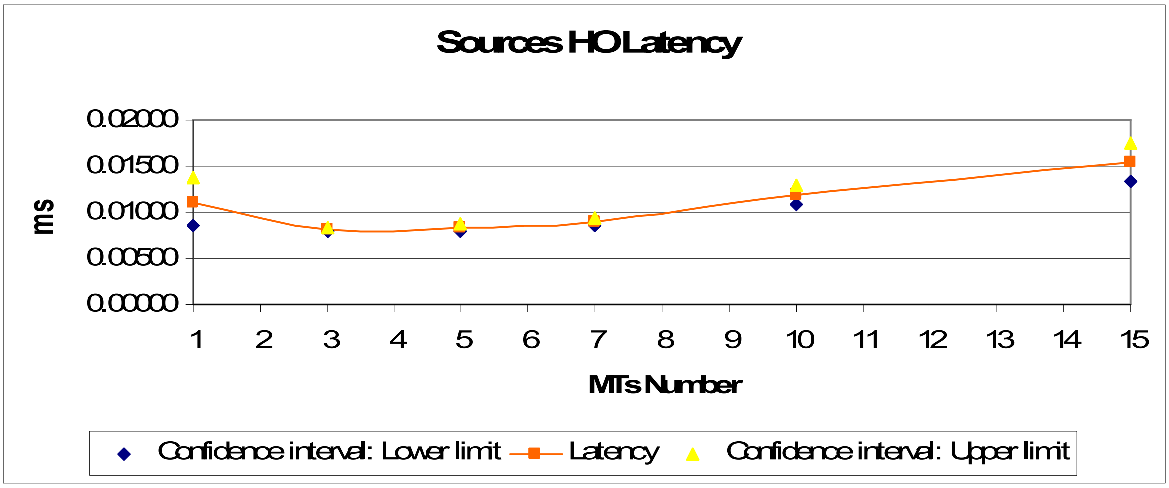

Figure 25 – Sources' HO Latency in Unicast Connections with MIP and NOAH

Except for the situation when the network has only one MT, the time needed by a source to complete the HO process increases in a very smooth way when increasing the number of MTs. The time needed for such process is around eight and fifteen milliseconds (8 – 15 ms).

However, the behaviour is much different when analysing listeners' Handovers. In the beginning of this thesis (Section 2.2 – Mobile IP), it was explained that when a MT changes its position, its IP address also change. A new IP address must be assigned to the host in order to route packets correctly to the host's new location. Due to dynamically managed IP-in-IP tunnels (in Mobile IPv4) and specially encoded packet-forwarding rules (in Mobile IPv6) overhead is introduced in the network. When the listener moves, its home agent tunnels the data packets in order to make them flow until the listener's new location. Using this strategy, the interval of time related to the HO latency are much larger than when it is a source movement. Increasing the number of mobile listeners values almost reach two seconds (2s) as is depicted in Figure 26 – Listeners' HO Latency in Unicast Connections with MIP and NOAH.

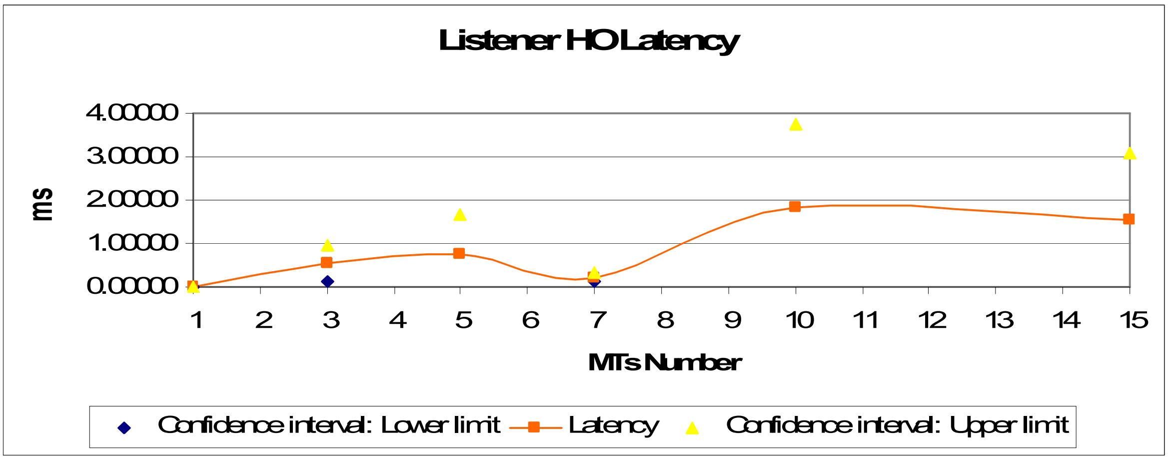

Figure 26 – Listeners' HO Latency in Unicast Connections with MIP and NOAH

5.3 – Evaluation of the MTAMM Solution

To evaluate the performance of MTAMM tests have been realized considering a network with six MTA domains and fifteen ARs fairly distributed. Sources and listeners randomly select the AR to connect. In Figure 27 – Evaluated Scenario, an example of one of the generated scenarios is presented.

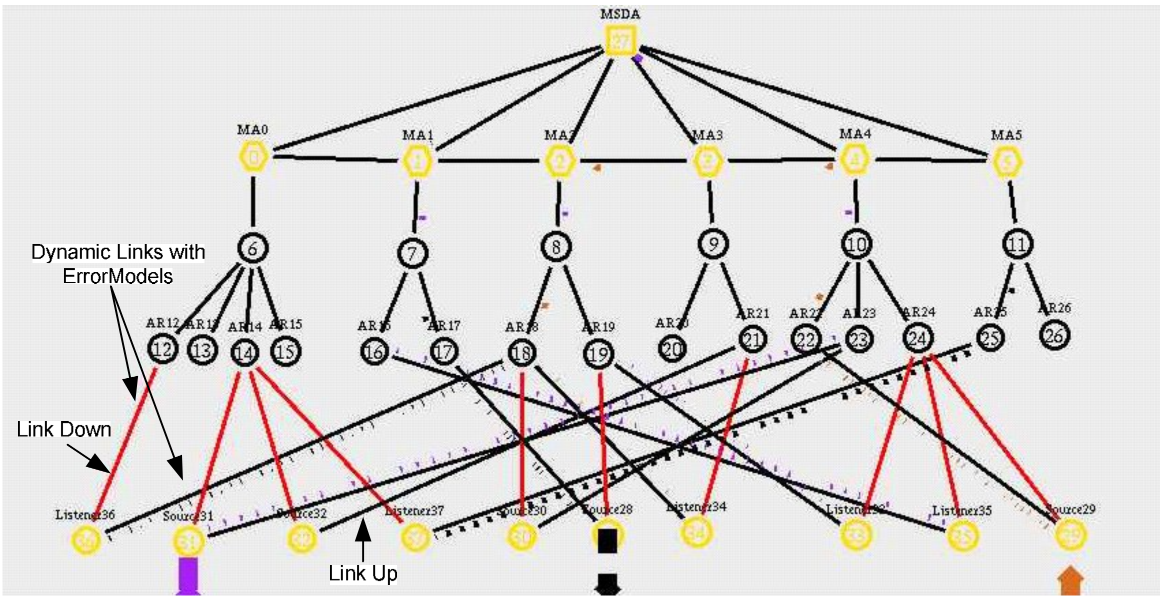

Figure 27 – Evaluated Scenario

Links between nodes in the core network (routers, MTAs, MSDA) have a bandwidth of 10Mbps and propagation delays of 2ms. The emulated wireless domain (wired dynamic links with ErrorModels) are configured automatically depending on the number of MTs (sources and listeners) according to values that were presented in Section 5.1 – Wireless Emulation, more properly in Table 8 – Configuration of ErrorModels. Each user is configured to act as a source or as a listener. The number of sources and listeners depends on the evaluated test. Each source only streams for one certain group and each listener receives from only one multicast group. This way, thee scenario will have as much multicast groups as sources. Sources and listeners are randomly placed in the network and their ARs are randomly selected. Each listener selects a randomly AR and a randomly selected group to start its request process and starts receiving the corresponding multicast data. This way, the network can have hundreds of multicast sources but only one multicast delivery tree if all listeners decide to select the same group. Of course, the probability for such situation is small when having multiple groups. Also, it is probable that there are several sources in the same MTA domain as well as several listeners that are receiving from the same group or from different ones. It is even possible to have the source for a certain group and its listeners in the same MTA domain. Resuming, we can have all combinations of sources and listeners in the same or different MTAs domains.

During the realization of the following tests, parameters have been changed while others remain constant along the simulation. The purpose is to test the influence in the performance and behaviour of the network. Five different tests were realized. They are now resumed and later more details for each one will be presented.

1. Influence of Sources Number

This test consists on the variation of sources' number maintaining all others parameters constant. We will consider five listeners, each one receiving only traffic from one multicast group. This way, with more sources we increase the probability of having more different multicast delivery trees. With one source we wave only one multicast tree and for example, with five sources, we will have at list five multicast trees, each one with exactly one source and one listener.

2. Influence of Listeners Number

The variation of listeners' number maintaining all others parameters constant will be analyse in this test. The number of multicast sources will remain unchanged (5 sources) and start sending data packets if one listener requests the correspondent group. This way, it is possible to have at list as much multicast delivery trees as listeners.

3. Influence of Multicast Traffic Rate

The purpose is to verify if multicast traffic rate influences network performance while maintaining other parameters constant. In this test the network has five (5) sources and five (5) listeners. Each listener randomly selects a multicast group and at this moment the corresponding source starts sending data packets. In this test, the maximum number of multicast trees is five if each listener selects one of the five available multicast groups.

4. Influence of Multicast Packets Size

Usually multimedia real-time applications use small packets, but this test aims to verify the network behaviour when packets tend to be bigger while maintaining other parameters constant. Like the previous test, this one also has five sources and five receives and also, at most, five multicast trees.

5. Influence of HO Frequency

In this test, the number of MTs that perform HO will vary while maintaining the duration of the simulation. This test is performed considering the same network with five sources and five listeners.

For each scenario (ten in each test) the simulation was repeated five times and the average values will be presented with the confidence values of 90%.

Test 1: Influence of Sources' Number

In this first test, and considering the network presented in Figure 27 – Evaluated Scenario, the number of sources is incremented from one to ten. Table 9 – Fixed parameters while testing the influence of Sources Number, presents the parameters that will remain constant along the several simulations.

Table 9 – Fixed parameters while testing the influence of Sources Number

MTAs	AR	CBR Rate	CBR packet size	Listeners
6	15	448 kbps	210 Bytes	5

This test considers a network with five listeners randomly distributed along the network. Thus, at list, five multicast delivery trees will be created if each one of these listeners selects one different group. Obviously, this is only possible when having five or more sources. For example, with one source there are only one available multicast group and all the listeners will request it. With more sources, it is probable that different listeners select different groups. Each source starts sending data packets when one listener, the first one, requests the corresponding group. Traffic from all sources has the same characteristics, CBR applications with a bit rate of 448 kbps and packets with 210 Bytes.

Figure 28 – Time Interval of SR Process while testing the influence of Sources Number, shows the time needed by a source to complete its registration process with a confidence interval of 90%.

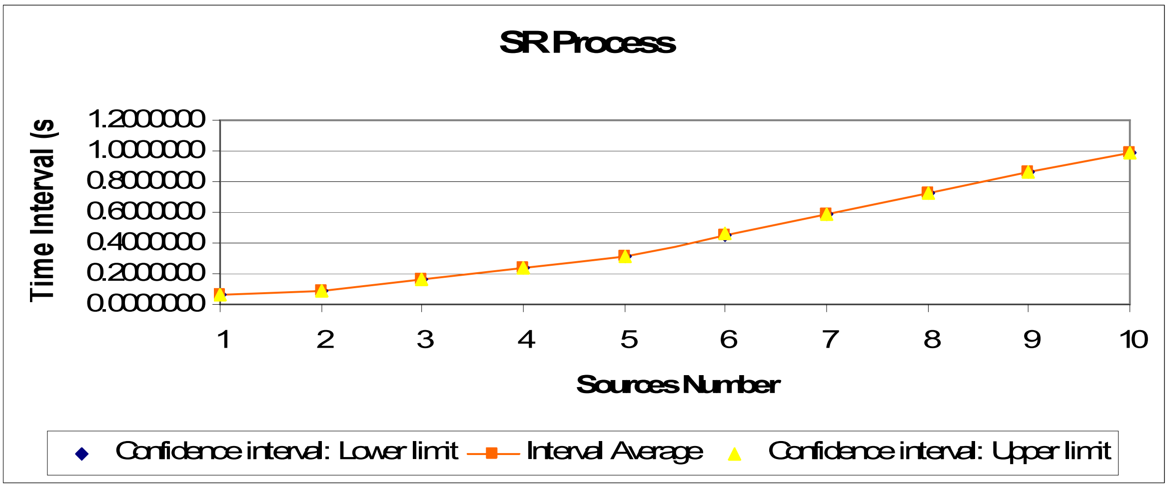

Figure 28 – Time Interval of SR Process while testing the influence of Sources Number

When the number of sources increases the time needed to complete the registration process also increases. This was the expected behaviour: when the number of sources increases, and consequently the number of MTs since the number of listeners is constant (5), the probability that collisions occur is larger. When the network has more MTs, more packets are dropped, delays between them and their access routers increase and the offered bandwidth is smaller which leads to the observed behaviour.

In Listeners Request process, the behaviour is similar as depicted in Figure 29 – Time Interval of LR Process while testing the influence of Sources Number.

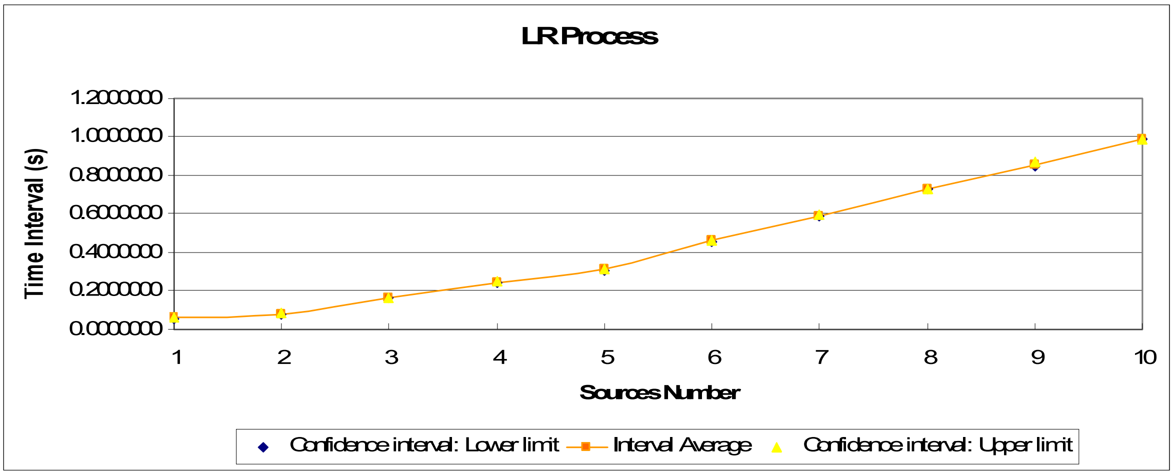

Figure 29 – Time Interval of LR Process while testing the influence of Sources Number

As expected, and for the same reason that was presented in SR process, as the number of sources increases also the time needed to accomplish the LR process increases. Comparing both processes, Source Registration and Listeners Request, they take approximately the same time which was predictable since the number of hops is the same.

When a source aims to realize an HO, it performs a new registration. The time needed to realize such a process is now illustrated in Figure 30 – Time Interval of SR HO Process while testing the influence of Sources Number.

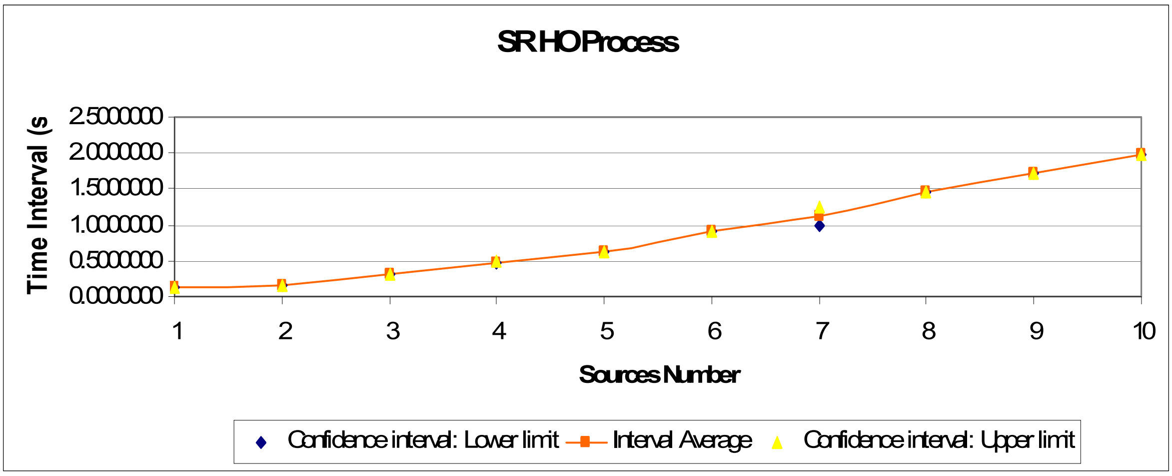

Figure 30 – Time Interval of SR HO Process while testing the influence of Sources Number

When a randomly selected source aims to perform an HO, this process takes more time for increased number of sources. More sources imply more MTs in the network and consequently more delays between them and ARs.

This HO process causes packet loss and as consequence, listeners will perceive traffic disruption until the multicast delivery tree for that source is rebuilt. This is presented in Figure 31 – HO Latency of SR HO Process while testing the influence of Sources Number.

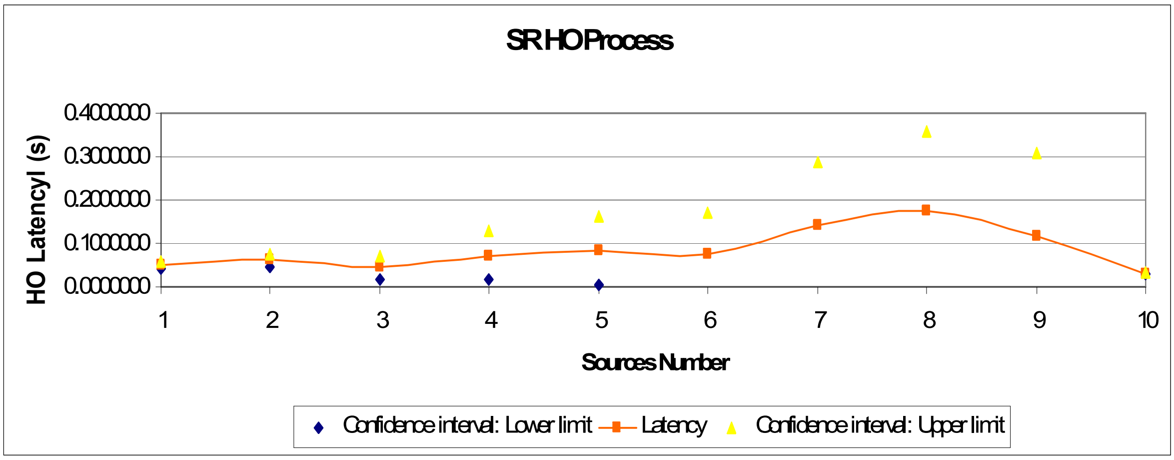

Figure 31 – HO Latency of SR HO Process while testing the influence of Sources Number

With more sources, the latency associated to the source HO tends to increase. Taking into account that the network has five listeners, at most, five multicast delivery trees are created. Thus, when having one source, there is only one tree with five different destinations and one source in common. With more sources, the probability is having more different trees. Note that, if the network has more trees, more traffic is sent and, consequently, the delay of the network is bigger making the HO process slower. Moreover, the disruption time that listeners feel, when the corresponding source moves, also depends on the type of HO that is realized. When a source moves, its multicast tree between him and its MTA has to be rebuilt and the time for such operation depends on the number of branch that have changed which are more in Inter MTA HO. This can explain why for some situations the HO Latency decreases.

When a listener aims to move it sends a LHO message in order to inform about the new position. The time needed to realize this process diverges depending if the HO is realized in the same MTA domain or for a different one. The time associated to this process is presented in Figure 32 – Time Interval of LHO Process while testing the influence of Sources Number.

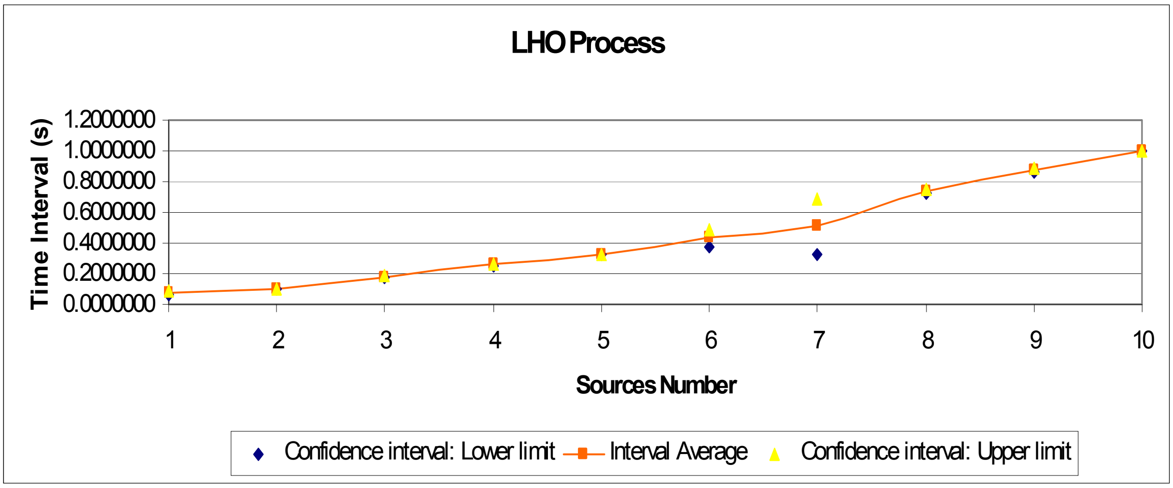

Figure 32 – Time Interval of LHO Process while testing the influence of Sources Number

When the number of sources increases, the time needed to accomplish the LHO process is higher. This is due to delays in routers queues, which are bigger due to more traffic, and also because delays in the wireless domain caused by collisions increases.

The disruption caused by the listener HO process is depicted in Figure 33 – HO Latency of LHO Process while testing the influence of Sources Number.

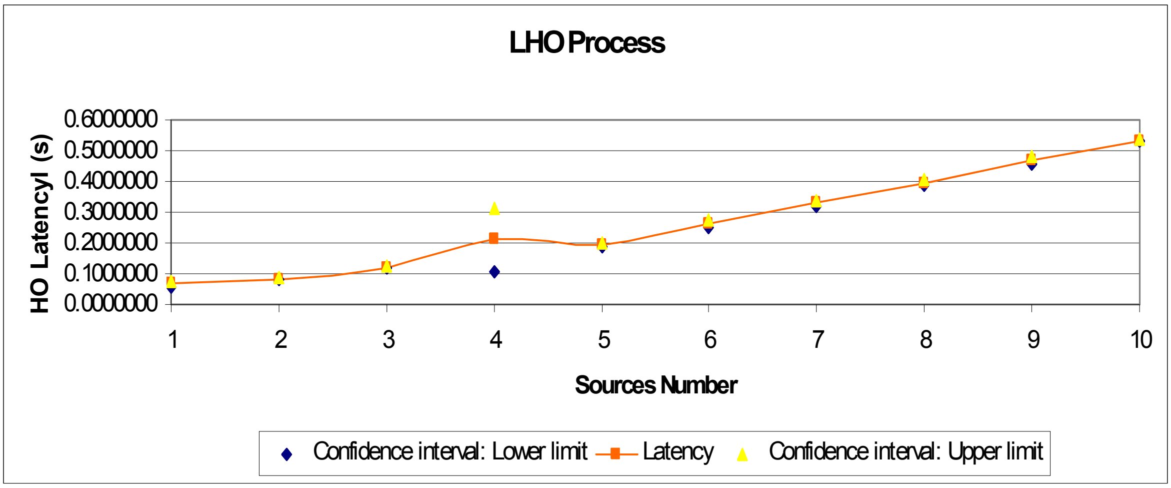

Figure 33 – HO Latency of LHO Process while testing the influence of Sources Number

More sources imply more traffic and more traffic leads to bigger delays. The time needed by listeners to perform the HO is influenced by network delay and by wireless delays.

Taking into account that listeners realize a make-before-break HO, so, in fact, when it moves, the multicast tree was already rebuilt.

Now that we have analysed the process that directly involves the mobile host, we will analyse the network performance. To perform all the processes that were presented, MTs and network agents need to exchange control messages. Those packets lead to the overhead presented in Figure 34 – Overhead while testing the influence of Sources Number

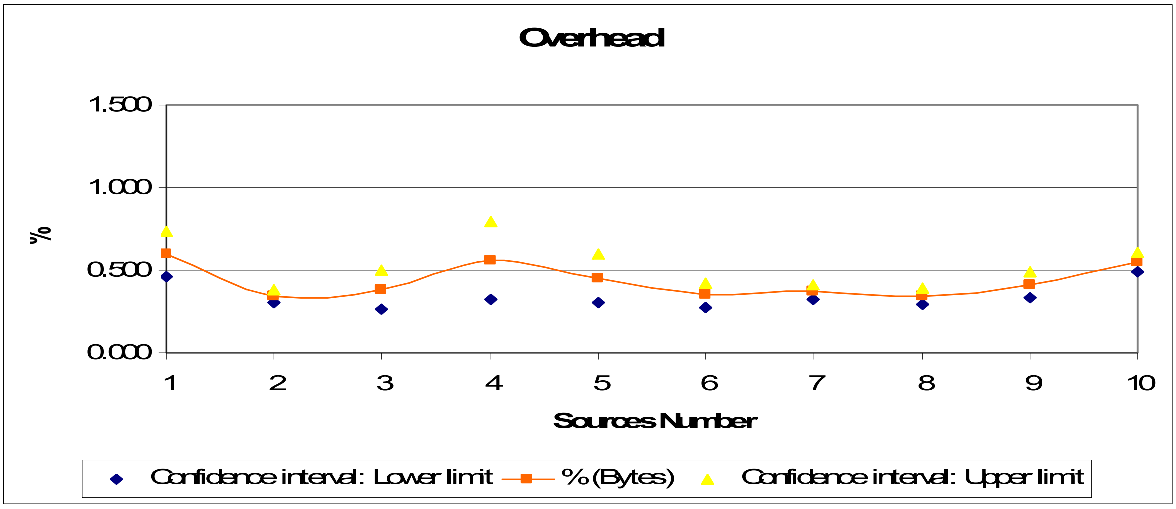

Figure 34 – Overhead while testing the influence of Sources Number

When having more sources more control messages are necessary. On the other hand, the number of packets in the network also increases because we have more traffic. This way, when increasing the number of sources, the traffic sent over the network also increases and the percentage of control messages relatively with all the traffic in the network is almost imperceptible, always smaller than 1%.

As consequence of the movements performed by sources and listeners and also due to the wireless domain more data is loss. This behaviour is depicted in Figure 35 – Percentage of packets loss while testing the influence of Sources Number

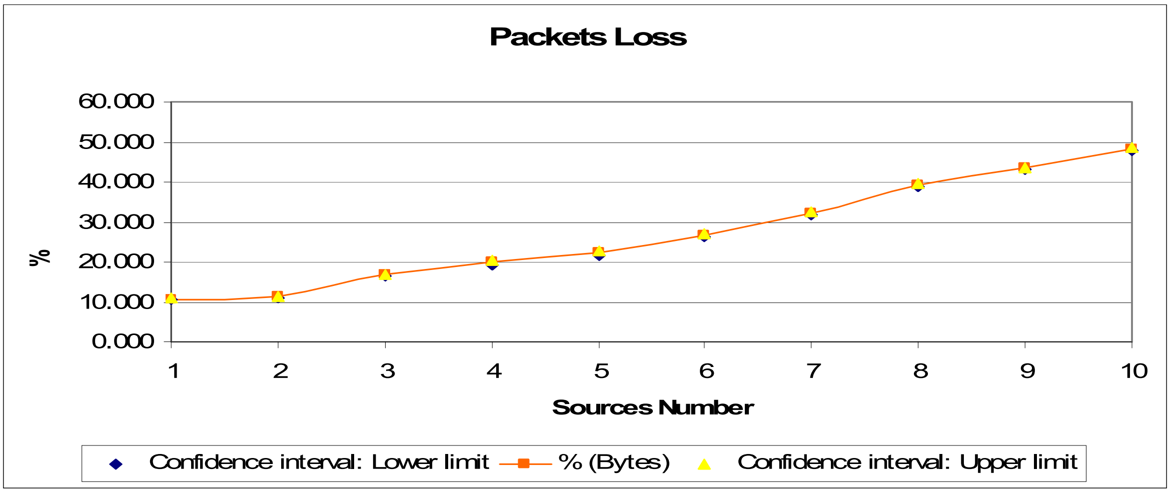

Figure 35 – Percentage of packets loss while testing the influence of Sources Number

When the number of sources increases, the percentage of Packets Loss is larger. This behaviour was expectable since with more sources we have more MTs and, consequently, more collisions occur in the wireless domain and more packets are dropped.

The delay on the network as well as the jitter are presented in Figure 36 – Delay while testing the influence of Sources Number, and Figure 37 – Jitter while testing the influence of Sources Number, respectively.

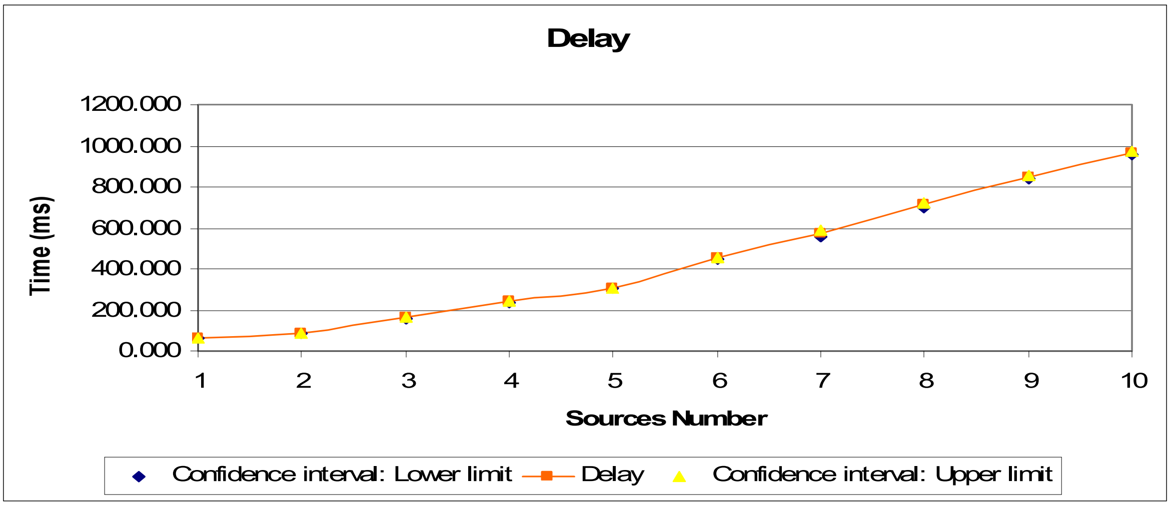

Figure 36 – Delay while testing the influence of Sources Number

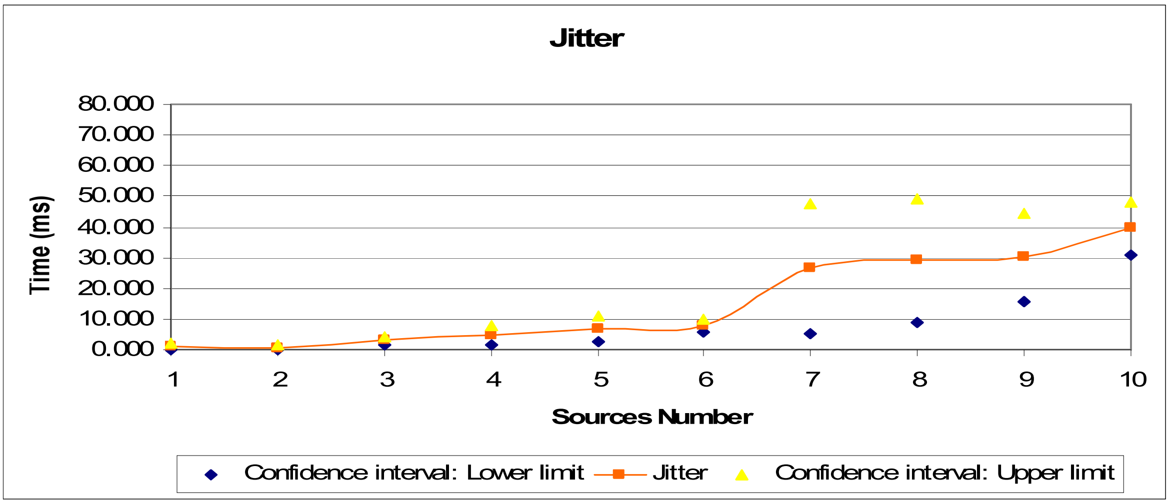

Figure 37 – Jitter while testing the influence of Sources Number

As expected, the delay on the network tends to increase when having more sources. In this situation, packets suffer a larger delay while travelling between mobile host and ARs, i.e., in the wireless domain. Also, since with more sources, we have more traffic, there are more packets in routers queues waiting for being forwarded. Also due to the presented reasons, the jitter is affected.

Test 2: Influence of Listeners' Number

In this second test, the parameter that was increased in each scenario is the number of listeners. The other parameters remain the same as presented in Table 10 – Fixed parameters while testing the influence of Listeners Number.

Table 10 – Fixed parameters while testing the influence of Listeners Number

MTAs	AR	CBR Rate	CBR packet size	Sources
6	15	448 kbps	210 Bytes	5

One different aspect from the last test is that, now, since the number of sources remains the same and it is the listeners number that is incremented, the number of delivery trees depends on the number of listeners. Sources are always ready to send traffic after they complete their registration process, but they only start sending when a listener request the corresponding multicast group. For example, in the first scenario we will have only one listener and five available groups. The listener will request only one of these groups and only one multicast tree will be built. With more listeners we have more probability to have several multicast trees. At most, we will have five different multicast trees but only if we have five or more listeners.

The SR and LR processes present a similar behaviour and are depicted in Figure 38 – Time Interval of SR Process while testing the influence of Listeners Number, and Figure 39 – Time Interval of LR Process while testing the influence of Listeners Number, respectively.

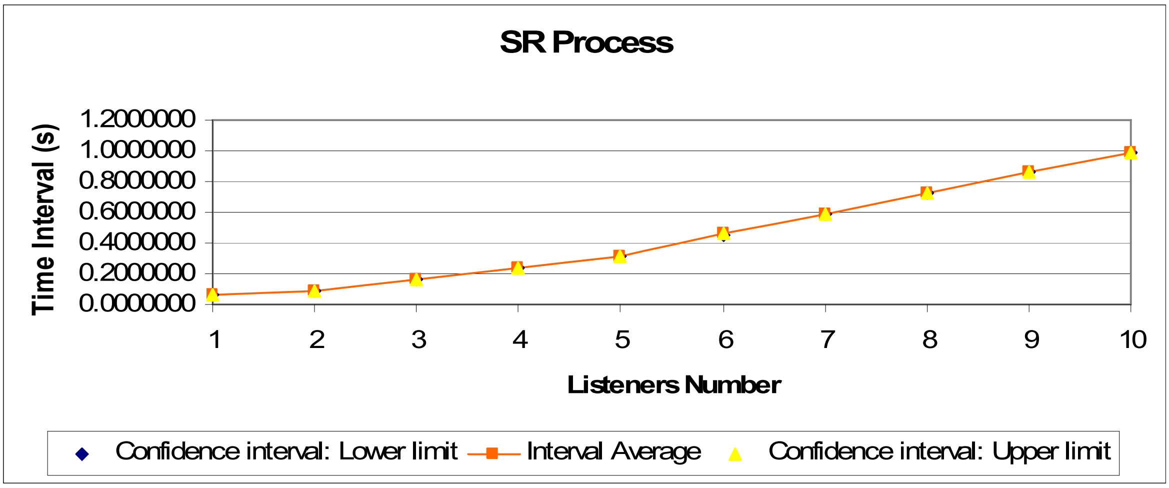

Figure 38 – Time Interval of SR Process while testing the influence of Listeners Number

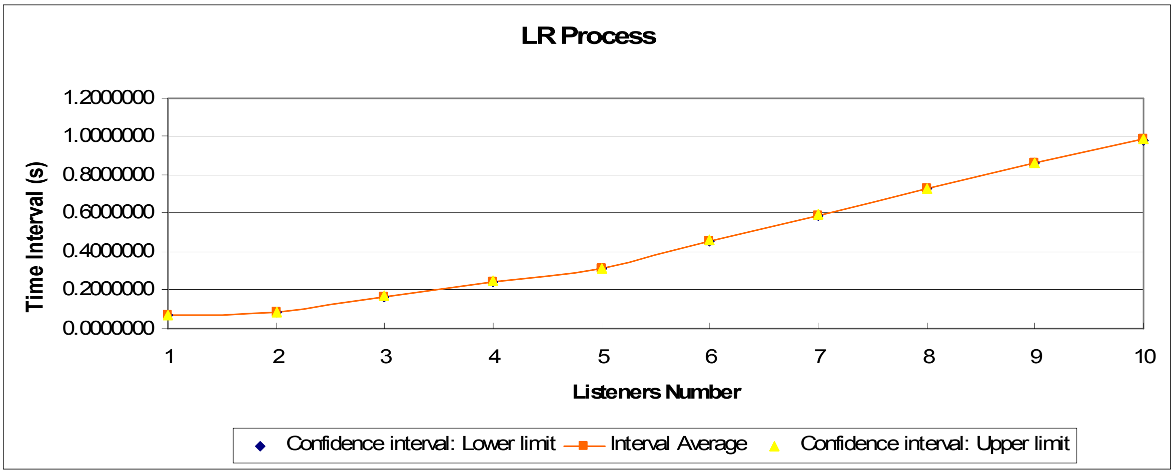

Figure 39 – Time Interval of LR Process while testing the influence of Listeners Number

When the number of listeners increases, both processes, the SR and LR, need more time to be completed. The reason for this behaviour is similar to the one presented in the first test, when the number of listeners increases, since the number of sources remains five, we will have more MTs and, consequently, the emulated wireless domain (wired dynamic links with ErrorModels) will offer less bandwidth, collisions will increase and packets loss and delays will be bigger. Furthermore, with more listeners we have a bigger probability to have more multicast trees, five at most. With more multicast trees, network has more traffic and, consequently, routers' queues will have more packets waiting to be forwarded. These reasons justify why both processes take more time to be accomplished.

When a source aims to move, a new registration is performed. This movement causes some disruption time from the listeners' point of view. These situations are illustrated through Figure 40 – Time Interval of SR HO Process while testing the influence of Listeners Number, and Figure 41 – HO Latency of SR HO Process while testing the influence of Listeners Number, respectively.

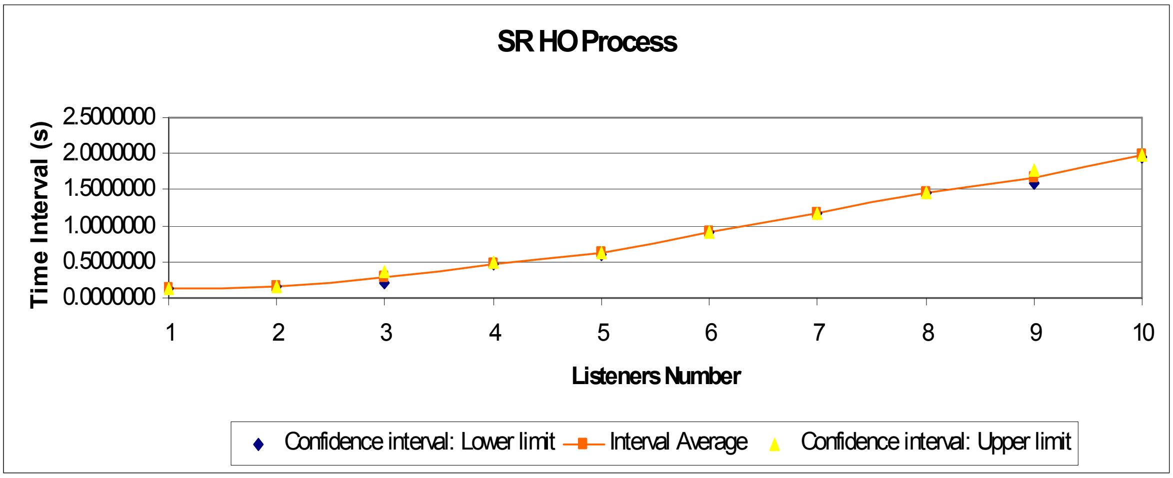

Figure 40 – Time Interval of SR HO Process while testing the influence of Listeners Number

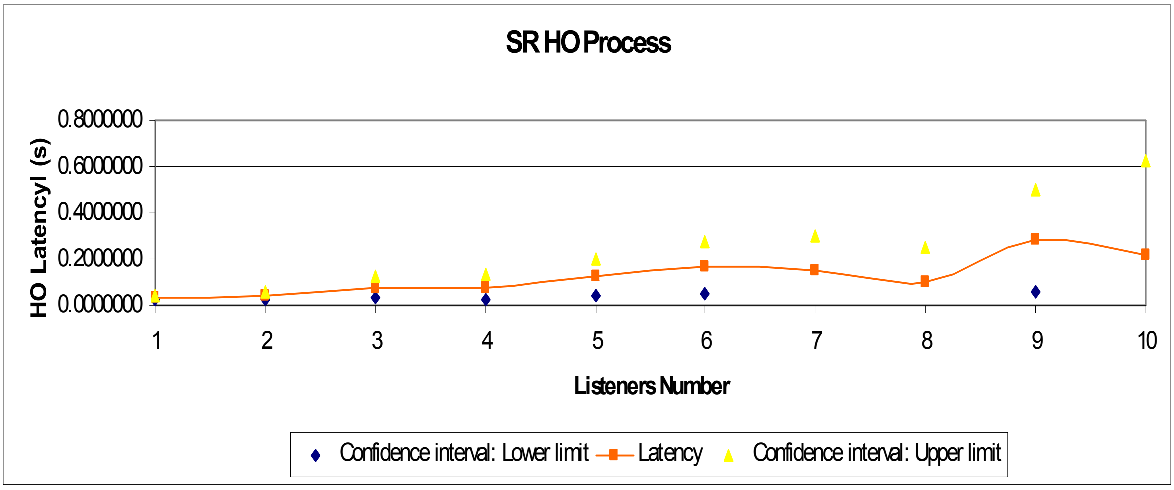

Figure 41 – HO Latency of SR HO Process while testing the influence of Listeners Number

The registration process that allows the source to perform an HO is affected by the number of listeners. More listeners imply more delays in routers queues and in the wireless domain. And also due to these reasons, packets take more time to reach their destination, so, listeners will percept a bigger disruption time. Some variations are due to the fact that sources can perform an Intra MTA HO or an Inter MTA HO. An inter MTA HO implies that the multicast delivery tree is more affected and more branches need to be rebuilt taking more time.

The LHO process and the caused disruption time are presented in Figure 42 – Time Interval of LHO Process while testing the influence of Listeners Number, and Figure 43 – HO Latency of LHO Process while testing the influence of Listeners Number, in that order.

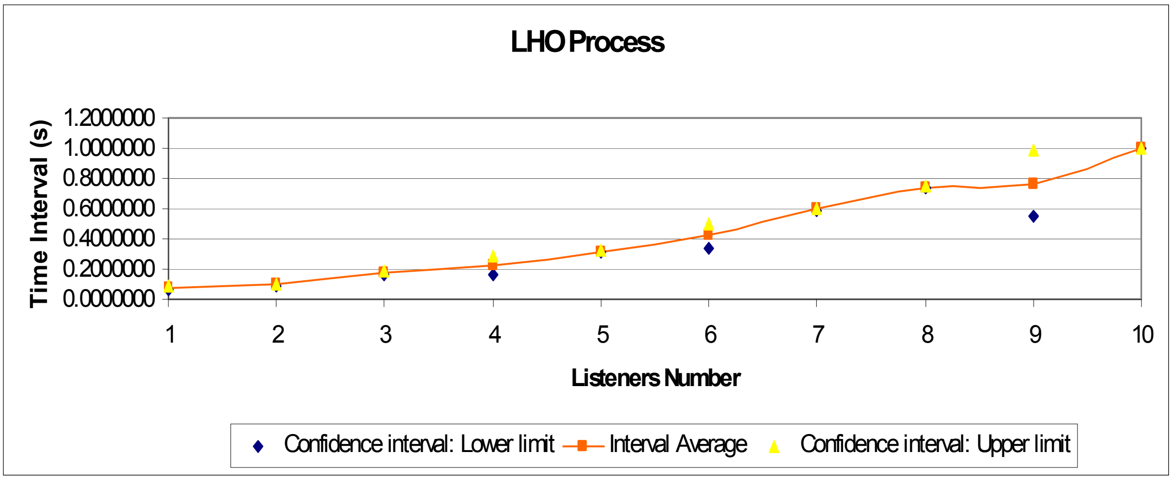

Figure 42 – Time Interval of LHO Process while testing the influence of Listeners Number

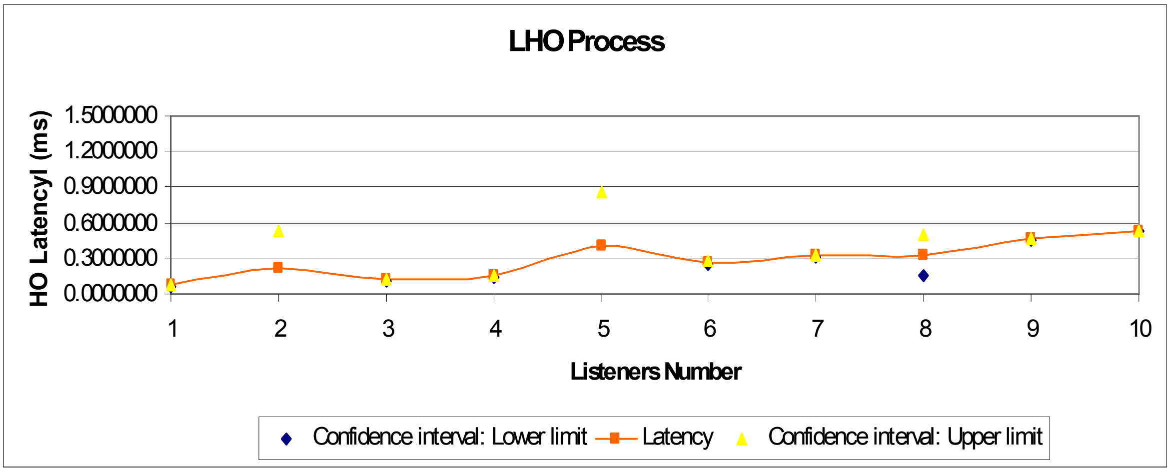

Figure 43 – HO Latency of LHO Process while testing the influence of Listeners Number

With more listeners network suffers from more delays and consequently also the LHO process need more time to be completed. The disruption time increases when the number of listeners is bigger, this difference is only due to delays that packets suffer in the emulated wireless domain. Because listeners perform a make-before-break HO the new AR is already receiving traffic when the HO is completed and the time needed to rebuild the tree is not taking into account in this measurement.

Figure 44 – Overhead while testing the influence of Listeners Number, shows the overhead caused by MDP messages. With more listeners the network need more control. However, the traffic increases much more so the percentage of control messages tends to be smaller.

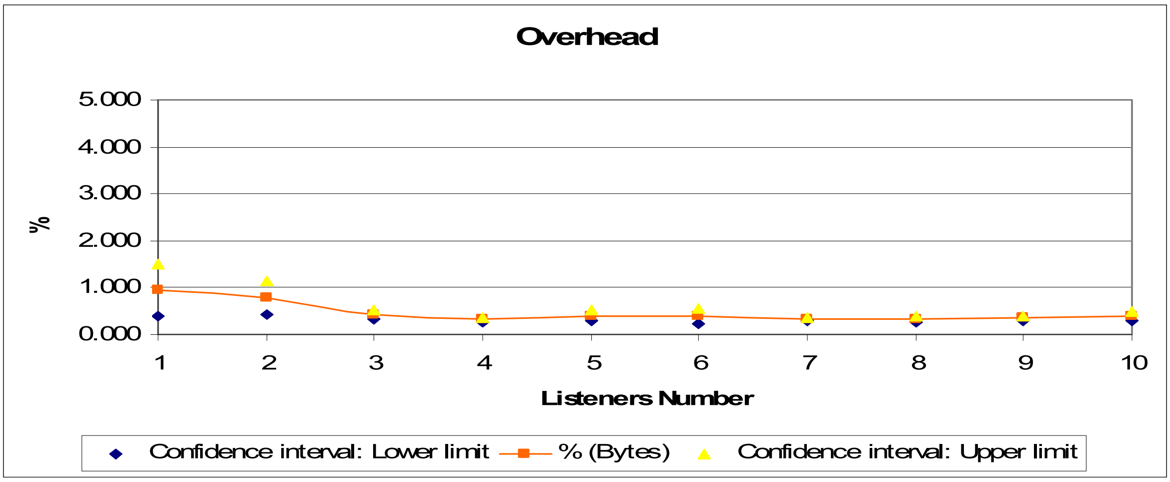

Figure 44 – Overhead while testing the influence of Listeners Number

With more listeners the emulated wireless domain causes more delays and more packets loss. This situation is presented in Figure 45 – Percentage of packets loss while testing the influence of Listeners Number, and in Figure 46 – Delay while testing the influence of Listeners Number, correspondingly.

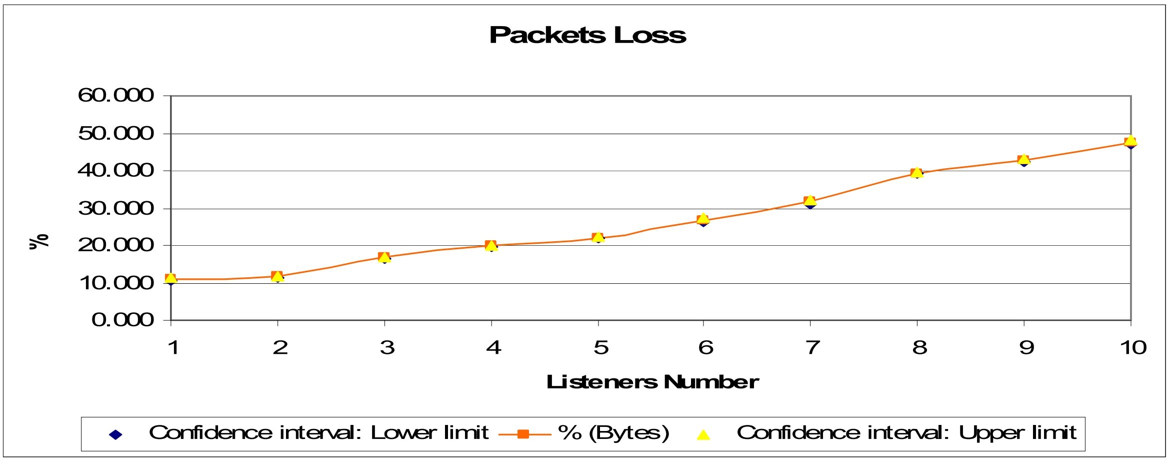

Figure 45 – Percentage of packets loss while testing the influence of Listeners Number

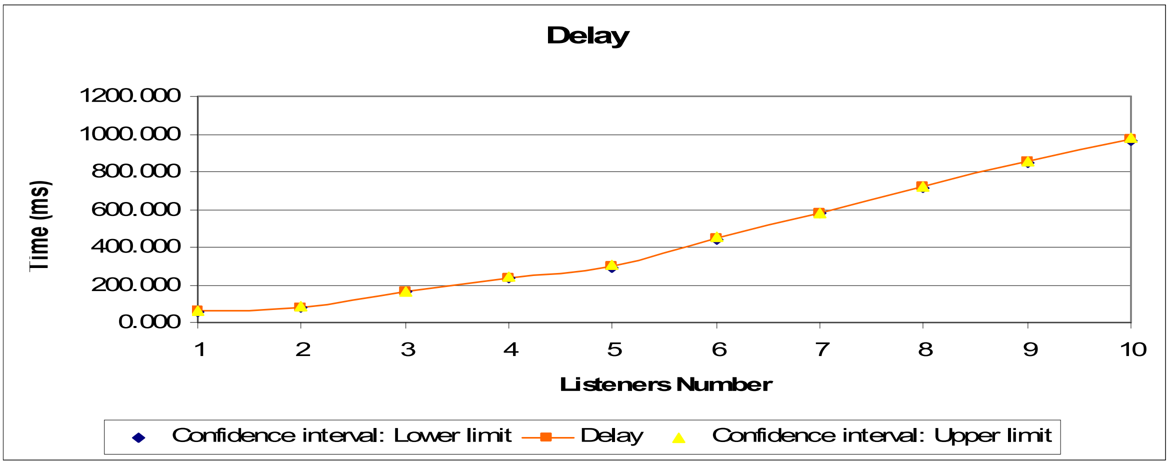

Figure 46 – Delay while testing the influence of Listeners Number

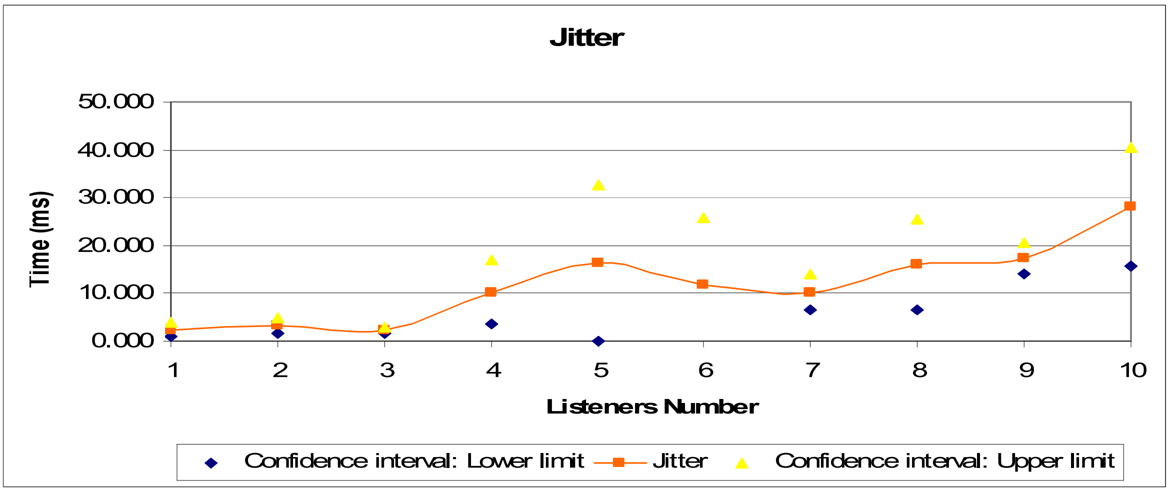

Figure 47 – Jitter while testing the influence of Listeners Number

Test 3: Influence of Multicast Traffic Rate

In this third test, the performance of the network was measured increasing the traffic rate. The number of sources and listeners remains the same (Table 11 – Fixed parameters while testing the influence of traffic rate).

Table 11 – Fixed parameters while testing the influence of traffic rate

MTAs	AR	Sources	CBR packet size	Listeners
6	15	5	210 Bytes	5

This test will have five sources, consequently five available multicast groups, and five listeners. Each one of these listeners selects one of the available group and starts receiving the corresponding data packets. Therefore, the maximum number of multicast trees is five if all listeners request one different group.

Results will now be analyzed. In this test, the time that a source needs to accomplish the SR process and a listener the LR process is not influenced by traffic rate. Both sources and listeners take around thirty milliseconds (30 ms) to accomplish their processes. Even if traffic rate is larger and, consequently, the network has more packets, MDP control messages have higher priority and they do not suffer larger delays with more traffic. When a source decides to move, it performs a new registration (SR) and, in this test, the time needed to perform such operation is not affected by traffic rate. Also with listeners, its LHO process takes the same time and do not depend on traffic rate.

The time that connections need to be re-established after HOs is depicted in Figure 48 – HO Latency of SR HO Process while testing the influence of traffic rate and Figure 49 – HO Latency of LHO Process while testing the influence of traffic rate.

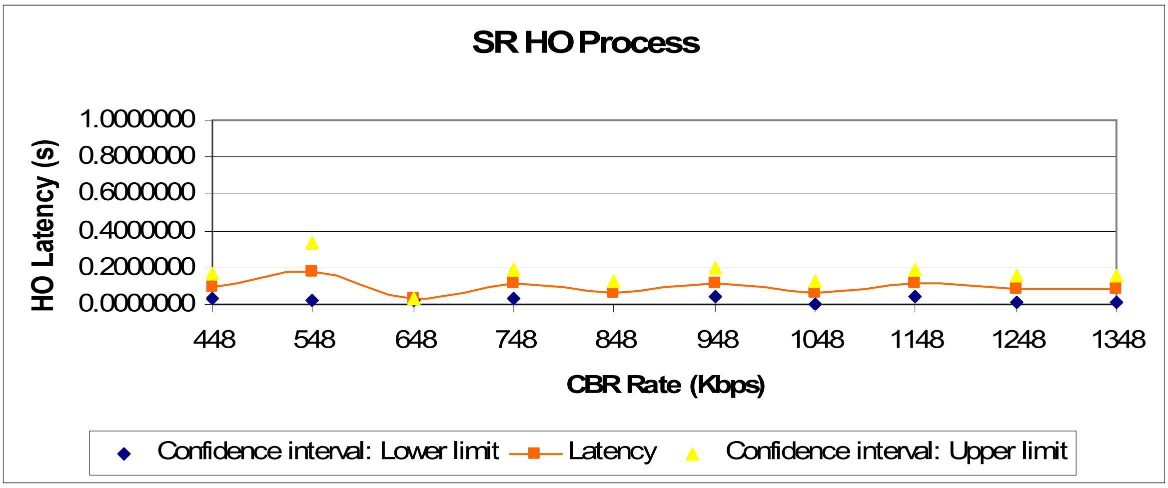

Figure 48 – HO Latency of SR HO Process while testing the influence of traffic rate

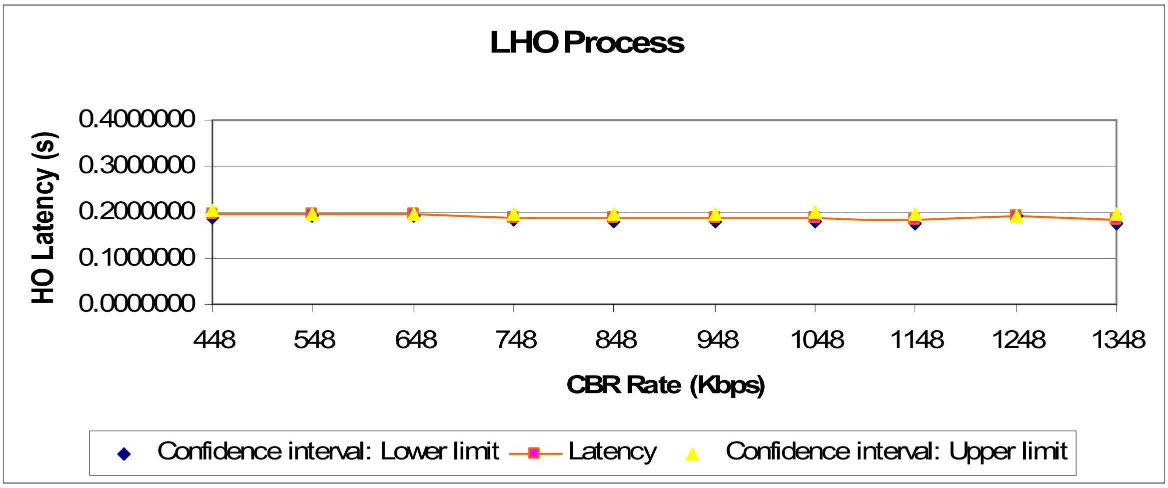

Figure 49 – HO Latency of LHO Process while testing the influence of traffic rate

The HO Latency time smoothly decreases with the increasing of traffic rate. However, the difference is practically imperceptible. With a larger traffic rate, delays between two consecutive packets are smaller and this is visible in these processes.

Attending to the overhead caused by MDP messages, Figure 50 – Overhead while testing the influence of traffic rate, shows that the percentage of the overhead decreases with the increase of traffic rate.

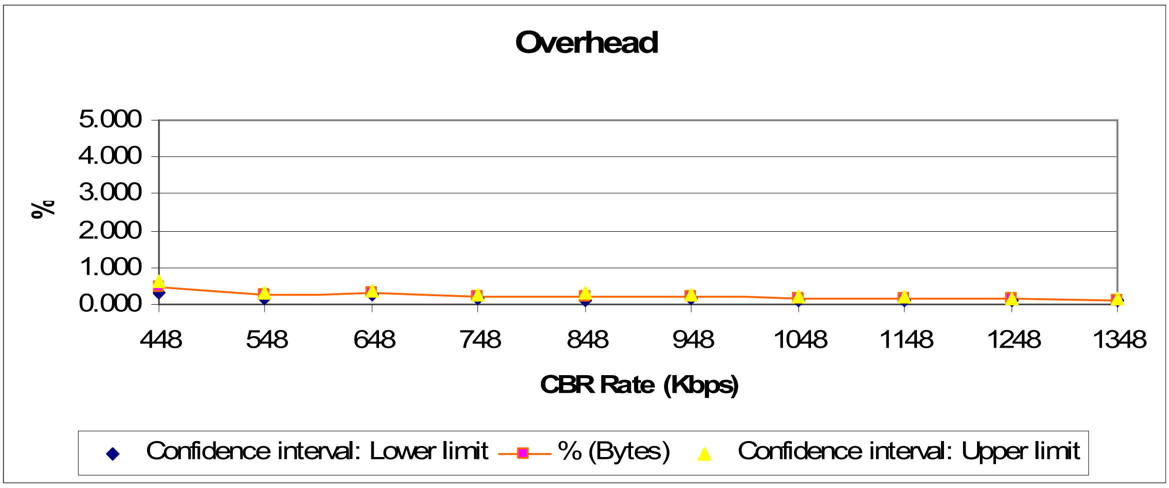

Figure 50 – Overhead while testing the influence of traffic rate

The Overhead decreases with the increase in the traffic rate but in a very smooth way. The number of bytes sent due to MDP messages does not obviously vary with traffic rate. Since in this test the number of MTs remains constant, the number of control messages remains practically the same. However, if the multicast traffic rate increases, the number of bytes sent also increase. In the same time interval, sources send more packets if the bit rate is larger and consequently the overhead caused by control messages becomes less evident.

In this test, the percentage of packets loss, delay and jitter is not affected by traffic rate and values remains almost constant.

Test 4: Influence of Multicast Packets Size

Test 4 was performed considering the increment of CBR packets size. The other parameters remain the same as in the above situations (Table 12 – Fixed parameters while testing the influence of packets size).

Table 12 – Fixed parameters while testing the influence of packets size

MTAs	AR	Sources	CBR Rate	Listeners
6	15	5	448 kbps	5

This test will consider five sources, consequently five available multicast groups, and five listeners.

Results are now presented. The time needed by a source to accomplish its registration process or by a listener to perform a request is not affected by Packet Size. When a source performs an HO, it makes a new registration (SR) and, in this test, the time needed for such process is not influenced by packets size. However, packets size has influence in the time needed to re-establish connections after the HO. This is presented in Figure 51 – HO Latency of SR HO Process while testing the influence of packet size.

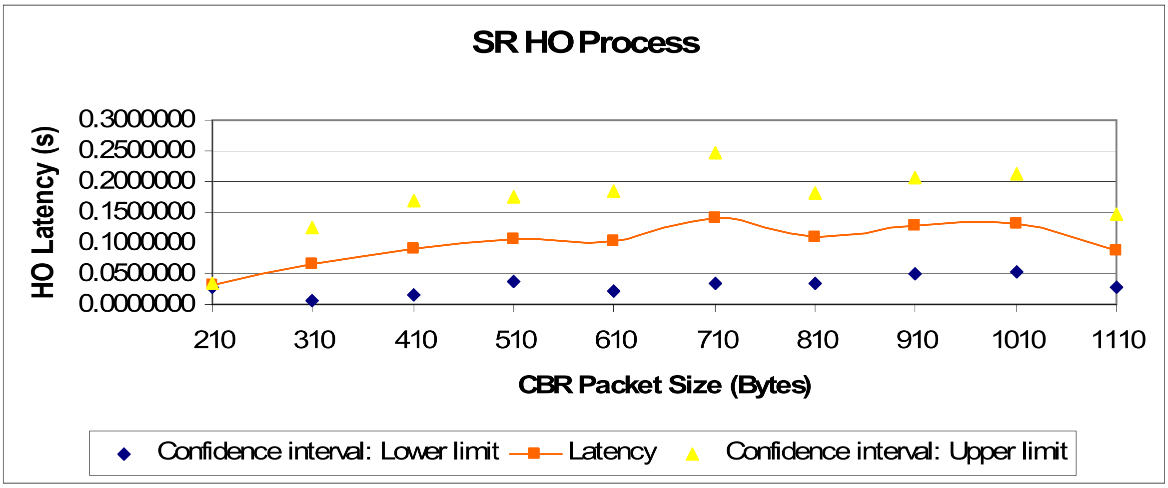

Figure 51 – HO Latency of SR HO Process while testing the influence of packet size

With larger packets, listeners have to wait more time until restarting the delivery of traffic. The reason is that larger packets suffer more delays in routers queues.

As with sources HO, also listeners HO (LHO) process does not depend on traffic packets size. Since listeners perform a make-before-break HO, the influence of packets size is not perceptible when moving from one AR to another one. The latency associated to the HO process remains constant.

CBR packet size does not influence substantially the overhead caused by control messages. The percentage of control packets is always smaller than one percent (1%).

Attending to the network packets loss, delay and jitter values are not affected.

Test 5: Influence of HO Frequency

In this last test, the performance of the network was measured increasing the number of HOs during the same duration of simulation (60s). The number of sources and listeners is constant (Table 13 – Fixed parameters while testing the influence of HO frequency).

Table 13 – Fixed parameters while testing the influence of HO frequency

MTAs	AR	Sources	CBR packet size	Listeners
6	15	5	210 Bytes	5

This test will has five sources and five listeners. Each one of these listeners selects one of the available group and starts receiving the corresponding data packets. Therefore, at most, five multicast trees are built if all listeners request one different group.

Results will now be analyzed. In this test, the time that a source needs to accomplish the SR process and a listener the LR process is not influenced by HO frequency.

In this particular test, when considering more HO, in some situations the SR process performed by a source in order to move to a different AR, or the LHO process by a listener were aborted since the same MT (source or listener) decides to perform a new HO before completing the previous one. In such situations, earlier sent messages do not make sense anymore, and the process is performed according to the last desired movement. The time needed to realize the new SR (HO) process as well as the time necessary to finalize the LHO process is not influenced by the HO frequency.

Figure 52 – Overhead while testing the influence of HO frequency, shows the overhead caused by MDP messages.

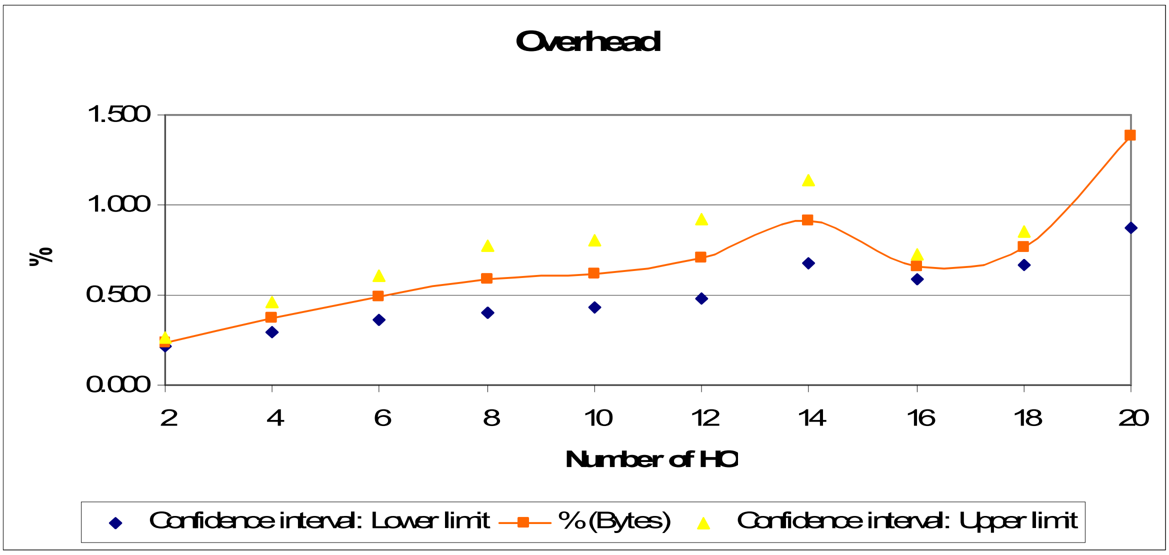

Figure 52 – Overhead while testing the influence of HO frequency

When considering more HO during the same period of time, the overhead caused by the MDP tends to increase. Such behaviour was expected since, in order to realize more HOs, both sources and listeners need to send more frequently MDP messages to perform the process. Taking into account that the number of sources and listeners is constant, traffic packets remains practically unchanged in all scenarios. Thus, the network overhead increases and becomes more perceptible reaching almost 1.5%.

Also the percentage of packets loss is affected by the number of HOs. When MTs perform more movements, more packets are lost. This kind of behaviour was predictable because, until the reconstruction of the multicast delivery trees and due to movements of MTs, packets are routed to the old location. The percentage of packets loss is presented in Figure 53 – Packets loss while testing the influence of HO frequency.

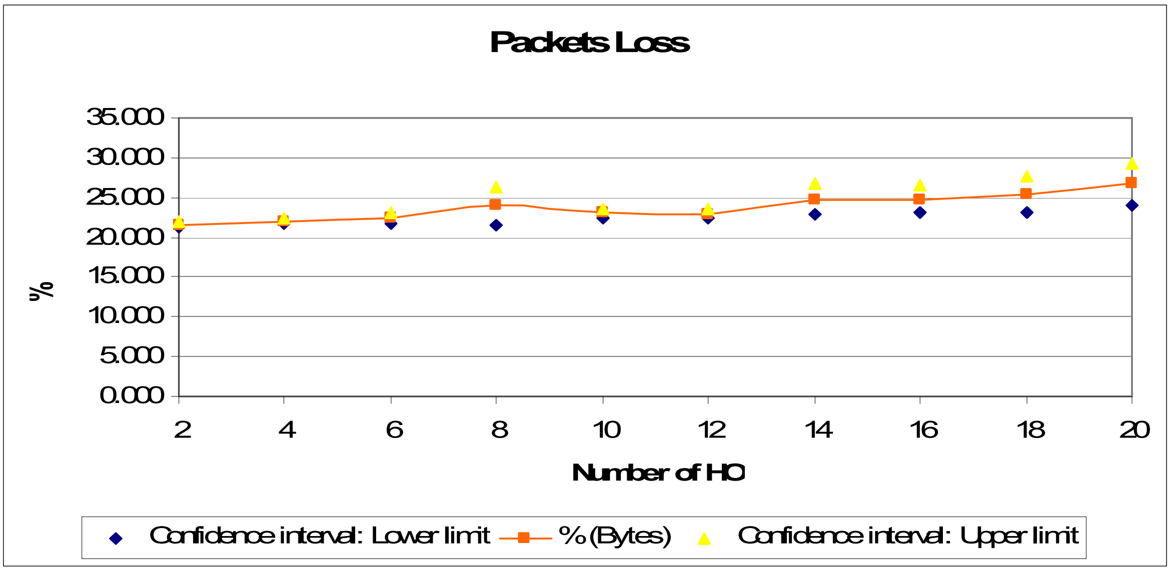

Figure 53 – Packets loss while testing the influence of HO frequency

The delay of the network was not affected with the variation of HO frequency. Values are in all scenarios around three hundred milliseconds. These values are similar to those presented in comparable scenarios where five sources and five listeners were considered. These delays are, essentially, caused by the wireless domain. However, the jitter is affected by MTs movements and as showed in Figure 54 – Jitter while testing the influence of HO frequency, tends to increase.

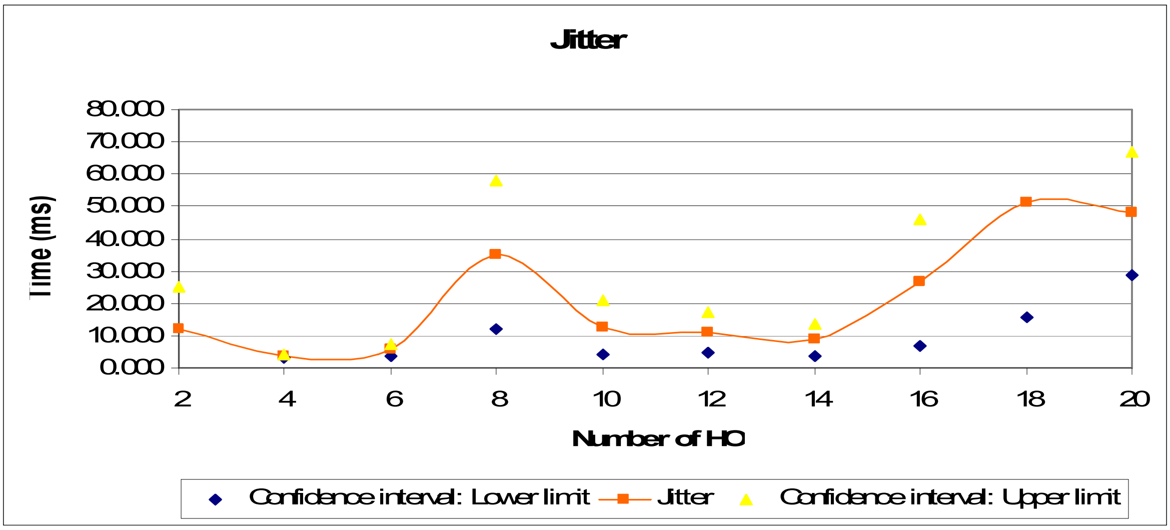

Figure 54 – Jitter while testing the influence of HO frequency

5.4 – Summary

This chapter presents the evaluation tests of MTAMM. Analysing obtained results allows concluding that MTAMM deals efficiently with multicast mobile terminals movements. The inherent architecture and protocol permit the division of the multicast delivery tree and, consequently, MTs move freely and the disruption times of sessions are minimized. Such behaviour is a crucial point for multimedia applications that require short delays. The evaluation of the MTAMM solution shows that this approach is perfectly considerable for seamless mobility in Next Generation Networks with multicast support.

The influence of the number of MTs, more properly, the sources' number as well as the listeners' number, was tested. More MTs cause more delays and, as expected, the network performance tends to decrease. Taking into account tests 3 and 4, traffic rate and packets size do not influence the time needed by sources and listeners to accomplish processes related to this mobility solution. However, these two parameters influence the latency associated to the HO and smoothly influence the network performance. The variation on the number of HO that occurs during the same period of time (test 5) influences the overhead of the network as well as the percentage of packets loss.

In the wireless unicast test with MIP and NOAH, sources and receivers movements have as consequence disruption time between 8 and 15 milliseconds for sources, and 8 milliseconds to 1.5 seconds for receivers. In the MTAMM solution, sources' movements have as consequence disruption time from 26 to 281 milliseconds. We can conclude that these values are larger. When analysing the disruptions time due to listeners' HO values, they are around 67 and 534 milliseconds. Since we are talking about multicast and multimedia applications, and taking into account that delays for multimedia applications are acceptable until 500 milliseconds, the MTAMM solution offers a proficient way in multicast mobility environments.

6 – Conclusions and Future Work

The purpose of this thesis was to develop and evaluate a new approach to multicast mobility environments. Although multicast is already mature and works efficiently in static networks, it is not yet deeply developed for mobility environments. Unfortunately, MIP does not explicitly take into account multicast mobility and only proposes two standard solutions: the Bi-directional Tunnelling approach – the mobile node tunnels all multicast data via its home agent; and the Remote Subscription strategy – mobile nodes always utilize their link-local addresses. However, these two solutions have some inconvenient and Agent-based solutions are being deeply studied.

This Thesis considered Agent-based solutions to evaluate a new solution for multicast mobility named Multicast Teleport Agent approach for Multicast Mobility (MTAMM). Multicast sources mobility is the bigger problem in multicast IP networks. So, in order to solve this issue, this approach considers static agents (MTAs and MSDA) that perform special functions and that use the MDP protocol to communicate between them and between them and MTs. One the one hand, the proposed solutions considers some agents to enhance the multicast mobility support. One example is the MSDA (Multicast Source Discover Agent), which is in charge of storing multicast sources' location. Another one is the MTA (Multicast Teleport Agent) that acts as a source and as a listener. It receives the data stream coming from the source and teleports it through a core channel to others MTAs. These MTAs, in turn, behave as a source from the point of view of listeners in their domain. This architecture aims to separate the multicast tree in two sub trees offering the possibility to MTs to freely move. This way, when a source moves, this action is not known by listeners. The proposed approach has presented the MDP (Multicast Discover Protocol) that offers a set of messages that allows MTs to perform some operations, and also allows network agents to communicate and support MTs mobility.

The MTAMM solution was developed and tested in a simulation environment using NS2. The referred agents and protocol had to be implemented. Some scenarios were created in order to test the efficiency of this solution and to extract some conclusions. These five tests were done considering a network with six MTAs and fifteen access routers fairly distributed.

In the first test, the influence of the number of sources was evaluated. With this test, we observed that the time needed by sources to complete their registration process, by listeners to perform a request and also time to accomplish HO processes, tends to increase with more sources. The reason is because, with more MTs, packets suffer larger delays in the wireless domain. We saw that in these conditions this architecture responds efficiently to HO situations and that the time needed to re-establish multicast connections is not very affected by the number of sources. When testing the influence of the number of listeners, the behaviour of registration and request processes, as well as in HO situations, is similar to the first test. We can conclude that with more MTs, those processes need more time to be accomplished, and that the variation of the number of MTs does not significantly affect the re-establishment time of connections after MTs movements. In both situations, more packets were lost with more MTS. The reason for such behaviour is that with more MTs (more sources in the first test and more listeners in the second one) implies more collisions in the emulated wireless network. Obviously, with more MTs, we have more traffic, and this is perceptible when looking to the network delay. In both tests, when the number of MTs increases, also the delay tends to increase. While testing the influence of traffic rate as well as packets size (tests 3 and 4, respectively), we can conclude that these parameters have no influence in the time needed to accomplish MTs processes (SR, LR, LHO). However, the packets size has influence in the re-establishment time of connections. Taking into account that larger packets suffer larger delays in the wireless domain as well as in routers queues, this is perceptible when MTs move. In both situations, the percentage of packets loss remains practically constant. The evaluation of the MTAMM solution shows that when MTs realize more HO affects mainly the network performance. In these five tests, we can conclude that the proposed protocol, MDP, allows controlling all the processes without introducing a significant overhead in the network.

We can conclude that MTAMM solution offers an efficient way to enable the full mobility of nodes in this environment. The presented architecture gives some abstraction to the terminals mobility. Since the multicast tree is not entirely centred in the source, when a MT moves, the reconstruction of the tree is faster and there is no need to re-establish current multicast connections.

However, further work is necessary. In order to improve the scalability of this approach, some features can be added. In the presented solution, each MTA is responsible for a domain. This way, all the multicast traffic from and to a MT in the domain has to flow through the MTA. This can cause serious delays in the MTA queue which is not desired for multimedia applications. In order to solve such problem, several MTAs can cooperate together in the same domain providing load balancing properties. Moreover, there are several types of multimedia applications and different kind of users. This way, the solution may provide quality of service. Traffic corresponding to a certain group and a certain application may have more priority than others. As consequence, characteristics such as packets loss and network delay should be better for these streams.

References

1 R. Hinden and S. Deering "Internet Protocol Version 6 Specification", RFC 2460, December 1998.

2 D.B. Johnson, C. Perkins and J. Arkko "Mobility Support in IPv6", RFC 3775, June 2004.

3 V. Devarapalli and F. Dupont "Mobile IPv6 Operation with IKEv2 and the Revised IPsec Architecture", RFC 4877, April 2007.

4 Romdhani, I., Kellil, M., Lach, H.-Y. et. al. "IP Mobile Multicast: Challenges and Solutions", IEEE Comm. Surveys, 6(1), 2004.

5 S. Deering "Host Extensions for IP Multicasting", RFC 1112, August 1989.

6 S. Bhattacharyya "An Overview of Source-Specific Multicast (SSM)", RFC 3569, July 2003.

7 H. Holbrook, B Cain "Source-Specific Multicast for IP", RFC 4607, August 2006.

8 D. Waitzman, C. Partridge, S.E. Deering "Distance Vector Multicast Routing Protocol", RFC 1075, November 1988.

9 D. Estrin, D. Farinacci, A. Helmy, D. Thaler, S. Deering, M.Handley, V. Jacobson, C. Liu, P. Sharma, L. Wei "Protocol Independent Multicast-Sparse Mode (PIM-SM): Protocol Specification", RFC 2362, June 1998.

10 B. Fenner, M. Handley, H. Holbrook, I. Kouvelas: "Protocol Independent Multicast - Sparse Mode PIM-SM): Protocol Specification (Revised)", RFC 4601, August 2006.

11 M. Handley, I. Kouvelas, T. Speakman, L. Vicisano "Bidirectional Protocol Independent Multicast (BIDIR-PIM)", RFC 5015, October 2007.

12 A. Ballardie "Core Based Trees (CBT version 2) Multicast Routing", RFC 2189, September 1997.

13 D. Thaler "Border Gateway Multicast Protocol (BGMP): Protocol Specification", RFC 3913, September 2004.

14 T. Bates et al. "Multiprotocol Extensions for BGP-4", RFC 4760, January 2007.

15 S. Deering, W. Fenner and B. Haberman "Multicast Listener Discovery (MLD) for IPv6", RFC 2710, October 1999.

16 R. Vida and L. Costa (Eds.) "Multicast Listener Discovery Version 2 (MLDv2) for IPv6", RFC 3810, June 2004.

17 http://tldp.org/HOWTO/Multicast-HOWTO-1.html#ss1.1

18 http://www.cisco.com/en/US/docs/internetworking/technology/handbook/IP-Multi.html#wp1020644

19 http://www.networksorcery.com/enp/protocol/mld.htm

20 Janneteau, C., Tian, Y., Csaba, S., Lohmar, T., Lach, Y.-H., Tafazolli, R., "Comparison of Three Approaches Towards Mobile Multicast", IST Mobile Summit 2003, Aveiro, Purtugal, 15-18 June 2003, http://www.mobilesummit2003.org, http://tools.ietf.org/html/draft-janneteau-nemo-multicast-mldproxy-00

21 V. Chikarmane, C.L. Williamson, R.B. Bunt, W. Mackrell, "Multicast Support for Mobile Hosts Using Mobile IP: Design Issues and Proposed Architecture," ACM/Baltzer Mobile Networks and Applications, 3(4):365-379, Jan. 1999.

22 T. Harrison, C.L. Williamson, W.L. Mackrell, and R.B. Bunt, "Mobile Multicast (MoM) Protocol: Multicast Support for Mobile Hosts," Proceedings ACM MOBICOM, Budapest, Hungary, September 1997.

23 J. Lai, W. Liao, M. Jiang, and C. Ke, "Mobile Multicast with Routing Optimization for Recipient Mobility," Proceedings IEEE ICC2001, pp. 1340-1344, June 2001.

24 K. Chen, N. Huang, and B. Li, "CTMS: A novel constrained tree migration scheme for multicast services in generic wireless systems," IEEE JSAC., 19:1998-2014, October 2001

25 C.L. Tan and S. Pink, "MobiCast: A Multicast Scheme for Wireless Networks," ACM MONET, 5(4):259-271, 2000.

26 Pedro Estrela, http://tagus.inescid.pt/~pestrela/cv/

27 Department of Informatics Engineering, University of Coimbra. http://eden.dei.uc.pt/~tandre/ssm_extension/index.htm.

28 Moy, J., "Multicast Extensions to OSPF", RFC 1584, Proteon, Inc., March 1994.

29 Perkins, C., "IP Mobility Support for IPv4", RFC 3344, August 2002.

30 J. Loughney, M. Nakhjiri, C. Perkins, R. Koodli, "Context Transfer Protocol (CXTP) ", RFC 4067, July 2008.

31 NS Manual, http://www.isi.edu/nsnam/ns/index.html

32 D. Thaler, B. Fenner and B. Quinn. "Socket Interface Extensions for Multicast Source Filters". IETF, RFC3678, Jan 2004.

Appendix A

IPv6

The IPv6 main header is required for every datagram. It contains addressing and control information that are used to manage the processing and routing of the datagram.

<table>
<tr><td>00</td><td>01</td><td>02</td><td>03</td><td>04</td><td>05</td><td>06</td><td>07</td><td>08</td><td>09</td><td>10</td><td>11</td><td>12</td><td>13</td><td>14</td><td>15</td><td>16</td><td>17</td><td>18</td><td>19</td><td>20</td><td>21</td><td>22</td><td>23</td><td>24</td><td>25</td><td>26</td><td>27</td><td>28</td><td>29</td><td>30</td><td>31</td></tr>
<tr><td colspan="4">Version</td><td colspan="8">Traffic Class</td><td colspan="20">Flow Label</td></tr>
<tr><td colspan="16">Payload Length</td><td colspan="8">Next Header</td><td colspan="8">Hop Limit</td></tr>
<tr><td colspan="32">Source address :::</td></tr>
<tr><td colspan="32">Destination address :::</td></tr>
<tr><td colspan="32">Data :::</td></tr>
</table>

Appendix B

IGMP messages

All IGMP messages are encapsulated in IPv4 packets, with an IP protocol number of 2, and are sent to multicast group addresses. Each packet is also sent with an IP *time-to-live (*TTL) of 1 and carries an IP Router Alert option (RFC-2113) in its IP heaher.

MAC header	IP header	IGMP packet

In IGMP version 3, the header has the following format

8		16		32 bit
Type		Max response time		Checksum
Group address				
RSV	S	QRV	QQIC	Number of Source
Source Address (1)				
...				
Source Address (N)				

MLD

The purpose of this protocol is to enable each IPv6 router to discover the presence of multicast listeners on its directly attached links and to determine specifically which multicast addresses are of interest to those nodes. All MLD messages are sent with a link-local IPv6 Source Address, an IPv6 Hop Limit of 1, and an IPv6 Router Alert option in a Hop-by-Hop Options header. The Router Alert option is necessary to cause routers to examine MLD messages sent to multicast addresses in which the routers themselves have no interest.

MAC header	IPv6 header	MLD header	Data :::

The header of MLD has the following format

00	01	02	03	04	05	06	07	08	09	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
Type								Code								Checksum															
Maximum Response Delay																reserved															
Multicast Address :::																															