Universidade de Aveiro
2007
Departamento de Electrónica, Telecomunicações e Informática

Rafael Figueiredo Sarrô

Avaliação de desempenho e Mobilidade em Redes Auto-Organizadas

Universidade de Aveiro
2007
Departamento de Electrónica, Telecomunicações e Informática

Rafael Figueiredo Sarrô

Avaliação de desempenho e Mobilidade em Redes Auto-Organizadas

Dissertação apresentada à Universidade de Aveiro para cumprimento dos requesitos necessários à obtenção do grau de Mestre em Engenharia de Computadores e Telemática, realizada sob a orientação científica da Prof. Dra. Susana Sargento, Professora do Departamento de Electrónica, Telecomunicações e Informática da Universidade de Aveiro

dedicatória Para a minha família, em especial para os meus pais.

o júri

presidente

Prof. Dr. José Neves
Professor Catedrático da Universidade de Aveiro

Prof. Dr. Mário Serafim Nunes
Professor Associado do Instituto Superior Técnico

Prof. Dra. Susana Sargento
Professora Auxiliar Convidada da Universidade de Aveiro

agradecimentos

Os meus sinceros agradecimentos à minha família, especialmente aos meus pais, aos meus irmãos e meus avós, pelo apoio e ajuda prestados.

Agradeço também aos meus colegas do Grupo de Redes Heterogéneas do Instituto de Telecomunicações, em especial ao Miguel e ao João Paulo pelo trabalho conjunto efectuado e também a todo o pessoal da "Sala de Bolseiros" pelos bons momentos que nos ajudam a passar o dia.

Uma palavra de apreço também para a Prof. Susana e para o Prof. Aguiar. Obrigado pela oportunidade oferecida, e por acreditarem em mim.

Especial agradecimento à música. A todas as bandas e artistas que durante a realização do meu trabalho me ajudaram a concentrar, a bloquear o excesso de ruído quando este era inoportuno e sobretudo por me dar força e ânimo para continuar quando a vontade era pouca.

Finalmente quero agradecer a todas as outras pessoas que não mencionei aqui, mas que de uma maneira ou de outra contribuíram positivamente para o meu trabalho.

A todos: Obrigado.

Palavras Chave

Redes Auto-organizadas, Avaliação Experimental, Mobilidade, Arquitecturas de Mobilidade, redes ad hoc

Resumo

As redes móveis ad hoc (ou auto-organizadas) são um assunto que nos últimos anos tem ganho muita atenção da comunidade científica. Os problemas associados a este tipo de redes foram amplamente estudados e expostos, foram propostas soluções, e algumas até foram tornadas um padrão da indústria. No entanto, a grande maioria do trabalho realizado, é dedicado a resolver só um problema de cada vez. Da mesma forma, as soluções que são testadas por forma a verificar a sua validade, muitas das vezes, são testadas recorrendo a trabalho de simulação. Uma parte do trabalho que é apresentado nesta dissertação de mestrado, junta uma série de protocolos desenvolvidos para redes ad hoc, os quais providenciam funcionalidades como: auto configuração, encaminhamento *unicast* e *multicast*, qualidade de serviço e taxação com incentivos numa única solução integrada que interliga as redes ad hoc a redes infra-estruturadas funcionando como uma extensão das mesmas. O demonstrador criado é avaliado de forma experimental, e os resultados obtidos são apresentados e discutidos. Uma vez que a rede ad hoc está interligada à rede infra-estruturada, num ambiente de quarta geração, é também apresentada uma arquitectura que suporta mobilidade de nós entre redes ad hoc e as redes infra-estuturadas que fazem parte do ambiente heterogéneo, e de este para as redes ad hoc é apresentada. A rede geral onde a rede ad hoc é integrada suporta novas tecnologias e tendências em gestão de mobilidade, tais como o protocolo em desenvolvimento *IEEE 802.21 Media Independent Handover* e gestão de mobilidade baseada em Domínios de Mobilidade Local. A forma como a rede ad hoc se integra com as tecnologias presentes na rede infra-estruturada, e como as especificidades da rede ad hoc são escondidas, é descrita e explicada.

Keywords Mobility, MANET, experimental evaluation, mobility architectures, ad hoc networks

Abstract Mobile Ad hoc network is a subject that has gained lots of attention from the research community in recent years. The problems inherent to this types of networks have been studied and exposed, solutions have been created and even standardized. However, the vast majority of the work performed is dedicated to only one problem at the time. In addition, the tests performed to validate the produced solutions are, most of the times, obtained through simulation work. The work presented in this thesis gathers together a set of ad hoc protocols, providing functionalities such as auto-configuration, unicast and multicast routing, quality of service and charging and rewarding in one integrated testbed, serving as a stub network in a hotspot scenario. A experimental evaluation is performed, and results are presented and discussed. Additionally, since the network belongs to a hotspot of fourth generation, a architecture that supports mobility of nodes between the ad hoc network and infrastructure networks is presented. The general network that includes ad hoc network integrates and supports the new technologies and tendencies in mobility management, such as the *IEEE 802.21 Media Independent Handover* and mobility management based on Local Mobility Domains. The way the MANET fully integrates with the infrastructure network, and how the ad hoc networks specific characteristics are hidden, is also presented and explained.

Contents

List of Figures

List of Tables

Acronyms

Acronym	Description
3GPP	Third Generation Partnership Project
AAAC	Authorization, Authentication, Accounting, and Charging
ADRM	Adaptive Demand-Driven Multicast Routing protocol
AIMD	Additive Increase Multiplicative Decrease
AODV	Ad-hoc On-demand Distance Vector routing protocol
AP	Access Point
AR	Access Router
CBR	Constant Bit Rate
CoA	Care-of Address
CPU	Central Processing Unit
CTS	Clear to Send
DNS	Domain Name System
DSR	Dynamic Source Routing
DYMO	Dynamic Manet On-Demand Ad hoc Routing
ECDSA	Elliptic Curve Digital signature Algorithm
GMD	Global Mobility Domain
GMP	Global Mobility Protocol
GSM	Global System for Mobile Communications
GW	Gateway
GW_INFO	Gateway Information
HA	Home Agent
HI	Host Identifier
HIP	Host Identity Protocol
HIT	Host Identity Tag
HMIP	Hierarchical Mobile IP
HoA	Home Address
IEEE	Institute of Electrical and Electronics Engineers
IETF	Internet Engineering Task Force
INRIA	Institut National de Recherche en Informatique et en Automatique
INSIGNIA	In-band Signaling system for supporting quality of service in ad-hoc networks
IP	Internet Protocol

Continued on next page

continued from previous page

Acronym	Description
IPv6	Internet Protocol version 6
LMD	Local Mobility Domain
LMP	Local Mobility Protocol
LSI	Local Scope Identifier
MAC	Medium Access Control
MAG	Mobility Access Gateway
MANET	Mobile Ad-hoc NETworks
MAODV	Multicast Ad-hoc On-demand Distance Vector
MGEN	The Multi Generator Tool set
MIG	Multicast Internet Gateway
MIH	Media Independent Handover
MIHF	MIH Function
MMARP	Multicast MAnet Routing Protocol
MN	Mobile Node
MOLSR	Multicast Optimized Link State Routing
MPR	Multi Point Relay
MSC	Message Sequence Chart
NEMO	Network Mobility
NIC	Network Interface Controller
ODMRP	On-Demand Multicast Routing Protocol
OLSR	Optimized Link State Routing Protocol
PACP	Polynomial-assisted Ad-hoc Charging Protocol
PDA	Personal Digital Assistant
QOLSR	Quality of service for OLSR
QoS	Quality of Service
RTS	Request to Sent
SCP	Secure Charging Protocol
SWAN	Service Differentiation in Stateless Wireless Ad-hoc Networks
TBRPF	Topology Dissemination Based on Reverse-Path Forwarding
TCP	Transport Control Protocol
UDP	User Datagram Protocol
UMTS	Universal Mobile Telecommunications System
VID	Virtual Identifier (or Virtual Identity)
WiFi	Wireless Fidelity
WLAN	Wireless Local Area Network

Chapter 1

Introduction

1.1 Self organized networks

Ad hoc is a Latin term which means ”for this purpose”. It is generally used to refer to a solution that has been designed for a specific problem, non-generalizable, and can not be adapted for other purposes. It is commonly used to describe a handcrafted work, a makeshift solution that appears to solve an inspected problem that requires a improvised resolution.

In computer networks, ad hoc is used to refer to spontaneous wireless network, that surges to establish a connection only valid for the duration of one session and requires no pre-established infrastructure. Instead, the participant devices discover each other within range of transmission to form a network for those computers. The vast majority of the networks referred to as ad hoc, uses the IEEE 802.11 standard operating in an *Independent Basic Service Set* (IBSS) mode.

Infrastructure networks rely on organization to accomplish communications. In these networks each element has a well defined role, and all elements are organized into complex architectures. The roles of the elements follow the client-server model, in where one element provides a service, and other(s) utilize that service. The entire architecture is designed and planned by engineers, and each element plays a well defined task on the architecture. The advantage of this level of organization is that the network administrator always knows the architecture and the role played by each of its constituting elements, easing the process of finding and correcting problems. If in small, low complex systems, this approach is good, when the size and complexity of the systems starts to grow (such as an large operator network), the management of the overall system starts to become a burden.

Ad hoc networks are self-organized, meaning that the roles of the elements of the network are not defined *a priori*, and that there is no need for or place to perform management task (apart from fixing hardware malfunctions). Current ad hoc networks already can perform many tasks independently: addressing can be performed dynamically and automatically; routing is capable of determine nodes location in the network, and react upon topology changes; network policies are automatically distributed to each node; service discovery mechanism aids nodes to detect available services in the network. However, self-organized networks are still very primitive and there are many situations in where more intelligence and automatic capabilities are desirable. The goal of self organized networks is to provide an intelligent network that can perform its task at the same time that promotes the management overhead to its bare minimum.

Ad hoc networks is a topic that has attracted much research from the academic community. This is due to the potential presented by the intrinsic concepts of self-organization and mutual cooperation towards a common goal. Although ad hoc concepts go back to 1972 in the Packet Radio Network (PRNET) project of the USA Department of Defense, it was only in the late 90s, when the IEEE gained interest in the concept and released the IEEE 802.11 [34] standard, which includes support for wireless ad hoc networks. A few years later, many hardware 802.11 capable was available in the market, allowing that more and more researchers could study and experiment the concepts of self organized networks. Nowadays, there is a plenitude of projects and research groups that include ad hoc networks as a topic of study. The IST founded European project like DAIDALOS [60], AMBIENT NETWORKS [59] and WIP [61] are examples of this kind of projects.

Despite all effort dedicated to ad hoc networks, there is still no clear definition to uniquely characterize this kind of networks. There are, however, four main types of networks that fall into the umbrella of ad hoc networks: Mobile, Hybrid, Vehicular and Sensor ad hoc networks.

Mobile Ad hoc Network (MANET) is the most common term associated with self-organized networks, being also the one that has the most commonly accepted definition. An Hybrid Ad hoc Network is an evolution from a MANET, and considers that the network contain special devices with special functionalities. These special nodes are used to interconnect the network to the internet. Another evolution from a MANET is a Vehicular Ad hoc Network. This type of network is characterized by the nodes being automobiles cruising a road that are able to communicate by means of MANET mechanisms. Finally a Sensor Network is a variety of an ad hoc network, where nodes (mobile or not), are very limited in terms of

computational power, battery life and radio range. This type of networks rarely use 802.11 as transport, but instead use ZigBee or Bluetooth.

1.2 Motivation

Nowadays users require to be connected to the Internet, accessing any type of service, anytime and anywhere. This requires that the access networks are always available, independently of the user's location, providing also the best access at every time. Currently deployed wireless access networks have limited coverage area, focusing mostly on high population areas. In this scenario, mobile ad hoc networks may provide the desired extension of the coverage areas, providing a more ubiquitous network environment. Also the MANET self-organization capabilities can be of great value for reducing the management overhead needed to maintain the network.

In these scenarios MANETs will be regarded as hotspot extensions, since they will extent, not only the area, but the availability of the hotspot. In these scenarios, MANETS need to be prepared to support the full set of services that the users can obtain from other access technologies. Users need to reach, and be reachable from, the Internet. They need to be able to communicate between them and with users across the Internet. In addition, users need the network to support quality of service to ensure that they get the contracted characteristics from the network. Operators providing access also need to be able to perform accounting and charge the users usage of the network.

Another functionality that mobile ad hoc networks, serving as hotspot extensions, need to support is the ability for the user to roam into, and out of, the network any time they need without loosing global connectivity or disrupting the active users sessions with its peers.

These scenarios, of ubiquitous and pervasive networks, frequently referred to as Fourth Generation networks (4G), applied to ad hoc networks, raise several research problems. Currently there is not a definitive solution for providing all the mentioned services in a reliably and effective manner, thus leaving many space for new solutions and architectures.

1.3 Objectives

The work to be performed for this thesis can be divided in two major parts. The experimental evaluation of an MANET demonstrator, and the development of a solution for integrate a

MANET in an heterogeneous mobility architecture.

In order to accomplish the experimental evaluation, a physical testbed has to be developed. Using the software developed during the DAIDALOS I project, a fully functional testbed will be constructed. The next step will be to design a series of test that are suitable to evaluate the performance of the testbed, according with the target scenario: a MANET functioning as a hotspot extension. The obtained results will be evaluated and discussed, from the point of view complete multiservice MANET.

The second part of the work will be to study the current trends in mobility in IP networks, and the way they are used in the DAIDALOS II project mobility architecture. The issues related to the inclusion of an MANET in the general mobility architecture of DAIDALOS II will be identified and a solution will be presented. The developed MANET architecture will be detailed and explained. Finally the remaining issues with the architecture will be identified, and classified for further research.

Summarizing, this thesis has the following objectives:

1. Study concepts associated with Mobile Ad hoc Networks, and its integration with infrastructure networks.
2. Construct and evaluate a demonstrator for the integration of the MANET.
3. Study IP networks mobility mechanism.
4. Study new paradigms for mobility in IP networks.
5. Develop a solution that provides the integration of ad hoc environments with the new paradigms in IP mobility.

1.4 Contributions

The following papers have been produced during the work performed for this thesis:

- Susana Sargento, Rafael Sarrô, Patrick Supar, Francisco Gallera, Marek Natkaniec, João Paulo Vilela, and João Barros. *Ubiquitous Access through the Integration of Mobile Ad hoc Networks*. 16th IST Mobile & Wireless Communications Summit, July 2007.

- Susana Sargento, Rafael Sarrô, Ricardo Duarte, Patrick Stupar, *Mobility in the Integration of Mobile Ad-hoc Networks*, STREP ENABLE and IP Daidalos MWCS Workshop - Research and Deployment Possibilities based on MIPv6, July 2007, Budapest, Hungary.

- Miguel Almeida, Susana Sargento, Rafael Sarrô, João Paulo Barraca, and Rui L. Aguiar. *On the limits of ad-hoc networks: Experimental evaluation*. Conftele - 6th Conference on Telecommunications, May 2007.

- Miguel Almeida, Rafael Sarrô, João Paulo Barraca, Susana Sargento, and Rui L. Aguiar. *Experimental evaluation of an integrated ad-hoc network*. 15th IST Mobile & Wireless Communications Summit, June 2006.

- Miguel Almeida, Rafael Sarrô, João Paulo Barraca, Susana Sargento, and Rui L. Aguiar. *Experimental evaluation of the usage of ad hoc networks as stubs for multiservice networks*. EURASIP Journal on Wireless Communications and Net-working, 2007:Article ID 62967, 14 pages, 2007. doi:10.1155/2007/62967

1.5 Disposition

This thesis is organized as follows:

Chapter 2 presents the mobile ad hoc networks, highlighting its special characteristics and requirements. In this chapter an overview of the solutions provided for ad hoc networks is presented. Unicast and multicast routing, auto-configuration, quality of service and charging and rewarding are briefly analyzed. In addition, this chapter analyzes these solutions and outlines the requirements for the MANET to operate as a hotspot extension.

Chapter 3 provides the experimental evaluation of the developed demonstrator. Among the various solutions proposed for ad hoc networks, the chosen solutions for integration into the demonstrator are presented. The developed testbed is described along with the set of tests performed for its evaluation. Finally, the obtained results are presented and analyzed.

In Chapter 4, the traditional mobility solution of IP based networks is presented an analyzed. The concept of Localized Mobility Management is introduced and described. The reminder of the chapter describes the mobility architecture (of the mobile node, ad hoc gateway and network) developed to support the inclusion of the MANET in a general mobility

architecture, that has support for new mobility protocols and paradigms. The MANET support for the used technologies is discussed an a final solution is presented. Finally, open issues are also presented.

Chapter 5 presents the main conclusions of this thesis.

Chapter 2

Mobile Ad Hoc Networks

Mobile Ad Hoc Networks (MANET), (along with vehicular, hybrid and sensor networks) are the most common types of self organized networks in present times. MANET however, is the most common term associated with self organization in today's literature, and is often used to refer to any kind of wireless ad hoc network. The IETF MANET Working Group [69], once defined a MANET as: "an autonomous system of mobile routers (and associated hosts) connected by wireless links — the union of which form an arbitrary graph. The routers are free to move randomly and organize themselves arbitrarily; thus, the network's wireless topology may change rapidly and unpredictably. Such a network may operate in a standalone fashion, or may be connected to the larger Internet". This definition was once part of the WG problem statement [26].

Research groups have focused their attention on MANET over the past years, and under the umbrella of the IETF MANET WG (in operation since 1997), several protocols have been proposed for solving the MANET problems (see next section). This momentum in research is due in part to the widespread usage and presence of personal computers equipped with IEEE 802.11 [34] compatible networks interfaces, which are capable of create, and participate in, ad hoc networks.

In this chapter the ad hoc networks specific characteristics (section 2.1) and special requirements (section 2.2), including a description of the various protocols that support them, will be presented. Finally section 2.3 describe the scenarios and requirements for an ad hoc network to serve as an hotspot extension.

2.1 Specific characteristics

Mobile Ad hoc Networks have several characteristics [32] that distinguish them from traditional networks. The most prominent characteristic is the lack of pre-established organization – MANETs are self-organized networks. The nodes inside the network must cooperate between them to achieve the basic services needed by the network.

The participants on a MANET are commonly referred to as mobile routers or nodes; a node is a host that integrates any wireless communication device. The nodes composing a MANET can be located anywhere, from airplanes to cars, and even on people or very small devices. A MANET can be regarded as an autonomous system of mobile nodes. This system may lay isolated, or may have gateways to a fixed network, so it can interface with it. When interconnected, the MANET operates as a *stub*[1] network connecting to a fixed network.

Depending on the nodes position and of their wireless device coverage area, nodes in the MANET form a random, multihop graph or ”ad hoc” network. This topology may change with time, depending of the movement of nodes. This multihop nature brings some side effects: Manets have highly dynamic topologies and are bandwidth and energy constrained and have limited physical security.

Dynamic topologies exist because nodes are free to move arbitrarily: as a consequence, the multihop network topology changes randomly and rapidly at undefined times. Also, nodes can have both bidirectional and unidirectional links. Due to the usage of wireless links, MANETs tend to have a limited bandwidth, significantly lower than wired networks. In addition, the characteristics of wireless communications (multiple access, fading, noise, interference, etc.) cause lower throughput than the maximum transmission rate. Energy constrain is also a problem for nodes in the MANET, since most of current mobile nodes are battery operated (PDA, cellphones, laptops, etc.) and power saving is becoming more and more an issue to be taken into consideration. Finally, mobile ad hoc networks are more prone to attacks to their security, than fixed wired networks. Since every node with a wireless network card can participate in the network, threats like eavesdropping, spoofing and denial-of-service attacks are more easily performed on MANETs.

[1] A stub network carries traffic originated at and/or destined to internal nodes, but does not permit traffic to transit through the network.

2.2 Specific requirements

There is a set of functionalities that need to exist in a MANET to enable it to function as a network, either dependent or independent of an infrastructure network. In order to be part of an ad hoc network, a node has to:

- Be identifiable. Each MANET participant must have an identifier so that other nodes can refer to it when they need to communicate. This identifier is usually an IP address of local or global scope. Global scope IP addresses are usually obtained by means of auto-configuration protocols.
- Be able to reach other nodes following an arbitrary path through the MANET. Nodes participating in a MANET must be able to route traffic sent by other nodes. This requirement is fulfilled by routing protocols (both unicast and multicast).
- Provide support for quality of service. The MANET network should be capable of differentiating between traffic flows, so that higher priority traffic can reach the desired destination according to the specified metrics (e.g. delay/jitter).
- Operate as a stub network. The MANET should be able to integrate with the services offered by the infrastructure network. The protocols used in the MANET need to interoperate with similar protocols in the infrastructure (see section 2.3).

The main requirement however, is that nodes must freely cooperate with its neighbors so that the MANET can function properly.

Each of these requirements will be satisfied by a dedicated protocol, or a combination of various protocols. Inside the MANET, the various protocols promote the interoperation between the nodes. Each protocol provides a service (or set of services) that will enhance the capabilities of the overall network. The solutions that fulfill the outlined requirements are described in the next sections.

2.2.1 Routing

In a multipath network, such as the Internet, the route that a packet must follow to travel to its destination is provided by the routing function of the network. Just like in the graph theory, a route is a path that describes the sequence of nodes that a packet must follow from

a starting point in the network (source) to a finishing point (destination). After the protocol has determined the route to the desired destination, the forwarding mechanisms process the packets according to the information in the routing tables, hop by hop.

MANETs differ from traditional routing in the fixed networks (of which the internet is a superset) because of its dynamic nature. In fixed networks the protocols determine a route during the protocol bootstrap, and then little or no changes need to be made to the routing tables. In a mobile environment, where the nodes change location frequently, the routes need to be constantly recalculated. Also, mobile ad hoc environments must cope with failures of the underlining IEEE 802.11 medium, unidirectional links, disruptive action or simply with selfish behavior perpetrated by malicious users. The topology of the network, and thus the route, may change during the lifetime of a session, requiring the routing protocol to update the routing tables between the end-points.

In the development of protocols that support the routing in the ad hoc networks, many different strategies and algorithms may be chosen. The routing protocols tend to be classified in different categories according to the solution chosen. This categorization helps choosing the best protocol for the target environment, including its weakness and strengths. Between the many categories that classify the routing protocols, the strategy used to determine best route between two end points is one of the most important. Other classification categories are the algorithm used to determine the route (link state or distance vector), the type of traffic to forward (unicast or multicast), how (or if) they handle geographical information, the effort they make in reducing the power consumption or if they create (or not) a hierarchical structure inside the network. The strategy used to determine routes can be separated in proactive or reactive. Proactive routing protocols, or table driven, constantly monitor network changes and update routes automatically. Reactive routing protocols, or on-demand, only maintain routes when some flow requires it. Additionally, there are some protocols that use both strategies, and therefore are denominated hybrids.

The proactive routing protocols maintain a distributed routing table among a set or all nodes in the network. This is accomplished by exchanging periodic messages between nodes. Using these messages, each node is able to build a representation of all, or part of, the network. The criteria to establish a route can be just the number of hops that keep apart two end points, or can be enriched with delay, jitter, security or even price information. The main drawback of the proactive routing protocols is the amount of control overhead produced due to the constant message exchange, even when there is no data to be delivered. This overhead usually increases with the number of nodes participating in the network. Since more control

messages will consume more resources (CPU time, wireless resources and battery time), proactive routing protocols tend to be inadequate for sensor networks, where nodes have low capabilities. In networks where routes have to be provided with low delay and constantly maintained, proactive routing protocols tend to be the best choice.

The main difference between proactive and reactive routing protocols, is that in the later, routes are not pre-calculated. When no packets are being routed, nodes have no information about the topology of the network or about any route towards any destination. Neighbors information is the only information gathered, since it can be obtained from the lower layer technology (IEEE 802.11). Routes are only established when there is data to be sent. At that time, the node floods the network searching for a route towards the destination. After the route is established, it is maintained as long as there are packets to be sent. When the data flow stops, and a given timeout expires, routes are dropped. Reactive protocols are more efficient in terms of network resources consumed. The computations needed to establish routes are simple, and only a minimal amount of information is kept. The control overhead only depends on the number of nodes in the network, their mobility and in the number of flows being forwarded. When no flows are present, control overhead is almost non-existent. The main drawbacks of these protocols are a longer latency establishing routes and lower route redundancy.

Following the separation based on the type of traffic to be routed, unicast an multicast routing protocols will be described in the next sections.

2.2.1.1 Unicast Routing

In the routing context, unicast refers to delivering packets to a single destination. By definition, unicast traffic is always sent from a single source to a single destination. Unicast traffic forms a point-to-point connection between two end hosts and in most cases represents a bidirectional data exchange. Unicast routing represents the vast majority of traffic delivered over today networks, in where a server and a client communicate directly using a dedicated (logical, not shared) connection.

Many ad hoc routing protocols were already proposed in the literature for unicast routing, commonly divided in proactive routing protocols, such as OLSR [31] and TBRPF [48], and reactive routing protocols, such as AODV [50], DSR [42] and DYMO [30].

AODV The Ad hoc On-demand Distance Vector [50] routing protocol belongs to the family of reactive routing protocols, meaning that it only provides routes when they are needed.

AODV algorithm provides dynamic, self-starting, multihop routing between nodes participating in an ad hoc network. It allows nodes to acquire routes quickly for new destinations, not requiring that nodes maintain routes to destinations that are not in active communication. Routes created by AODV are loop free and quickly restored upon link breakage. When a route breaks, AODV notifies the affected set of nodes so that they invalidate and repair the route quickly.

When a route to a destination is needed, AODV propagates through the network a *Route Request* (RREQ) message. The node that sends a RREQ is known as the originator. This request is efficiently forwarded in the network, avoiding loops (and retransmission) through the usage of a distinct sequence number in each request. When a node receives a RREQ message and has knowledge of a route to the destination, it issues a *Route Replay* (RREP) message to the origin of the RREQ. In order for the RREP can be sent in unicast to the originator, each router that propagates a RREQ caches a route back, registering the neighbor address from which the RREQ was sent. RREP are then forwarded hop by hop to the originator, making the route active in each node that forwards it. The originator can receive multiple RREP messages, when there are multiple paths to the destination, and is capable of choosing the most suitable route based on the decision metrics (usually the number of hops). Routes are valid only when packets for that route are being forwarded. If no packets are forwarded after a timeout, the route is discarded. This mechanism helps reducing the old (and often incorrect) route information in the routing tables. After a route is established, all nodes monitor their link with the neighbors in the path, using driver information, or recurring to special RREP messages called *Hello* (HELLO). When a link breaks, due to topology changes or node malfunction, the node detecting the break will issue a *Route Error* (RERR) message towards the sender. This message will invalidate the current route in all nodes that receive it. After the originator receives a RERR message, it must repeat a route discovery process to determine a new route.

There are several public implementations for the AODV protocol [63], both for simulation environments as for real-life experimentation, supporting different operating systems, architecture and IP versions. Also, inter-operation between these versions have bewn tested [3] with success.

OLSR The Optimized Link State (OLSR) routing protocol is a proactive routing protocol. OLSR is one of the most mature and popular routing protocols for MANET. It was first developed at INRIA and first proposed in [7]. Later it was developed inside the IETF MANET [69] charter, being proposed as an experimental RFC status in RFC 3626 [31]. As AODV, there are many public implementations of OLSR, many of which conform with RFC 3626 and support most operating systems.

Various OLSR messages can be encapsulated and transported in one packet, optimizing the network resources required for protocol operation. Each message starts with a header identifying the message type, so that nodes can identify quickly if they can process, and forward the messages. This behavior is useful, since not all nodes support all messages. In addition, OLSR gains flexibility, since new functionalities can be introduced not requiring that old nodes have support for it.

OLSR operation is divided in 3 main functions: neighbor sensing, optimized flooding and forwarding, and link-state messaging and route calculation. The neighbor sensing function provides information about the status of links between nodes. It is implemented by the exchange of periodic *HELLO* messages. This procedure can determine, and use, both unidirectional and bidirectional links, and is of great importance to determine and maintain routes during topology changes.

The optimized flooding and forwarding function is used to efficiently spread the messages across the network and to the determination of a forwarding route. OLSR employs the method of Multi Point Relays (MPR), in order to reduce overhead by avoiding retransmissions. With MPR, the flooding process is directed through a distribution tree. MPR tree varies with the topology of the network and the number of nodes. In addition, nodes may choose not to participate in the forwarding tree and can choose to participate in the tree of other MPRs. To chose the MPR, each node calculates the best neighbor which allows reaching any node two hops away. Redundant nodes are detected and removed resulting in a tree that provides close to optimal routes.

The link state messages are only sent by MPRs, and only MPRs are included in the messages. This characteristic allows an efficient reduction in the complexity of route computation and the overhead of the routing protocol.

By design, OLSR has native support for multi-homed nodes, and disseminates interface information to other nodes in the network. This facilitates the process of interconnecting the ad hoc network with other kinds of networks.

As a consequence of being a proactive protocol, OLSR has a high overhead in networks with high number of nodes, much larger than the overhead of reactive solutions, which makes it unsuitable for low power environments such as sensor network. However, OLSR tends to perform better in large ad hoc networks. Contrary to reactive routing protocols, that significantly increase the overhead with the number os nodes, OLSR maintains a constant grow of the overhead, causing it to have less overhead.

2.2.1.2 Multicast Routing

Multicast is the delivery of packets to a group of destinations simultaneously. To accomplish that, it uses efficient forwarding techniques that only require that each packet travels over each link only once. Multicast routing creates a distribution tree, rooted at the sender, and with the leafs on the receivers. In order for a node to receive a multicast data flow, it needs to explicitly join the distribution tree, so that traffic can be delivered to it.

Multicast is most useful for multimedia services, such as IP Television. These services are normally based on membership rules, and the same content is distributed to a large number of clients. Multicast routing is able to deliver the same content to multiple clients upon proper service subscription. The cost to the network is some additional signaling required to maintain the distribution tree and client subscriptions. However, the load on the network as the number of clients increase is close to O(1), instead of the typical O(N) presented by unicast.

For MANET there are several protocols that provide efficient multicast routing. MAODV [52] and MOLSR [36] are, respectively, the multicast versions of AODV and OLSR. On-demand Multicast Routing Protocol (ODMRP) [57] and Adaptive Demand-Driven Multicast Routing (ADMR) [40] are multicast ad-hoc routing proposals that reduce the overhead of maintenance of the multicast tree in the ad-hoc network. The Multicast MAnet Routing Protocol (MMARP) [17] provides a proper integration with infrastructure networks, besides multicast routing inside the MANET.

MMARP The Multicast MAnet Routing Protocol (MMARP) is especially designed for MANETs, and is fully compatible with the standard IP multicast model. Since MMARP supports the IGMP [29] and MLD [54] protocols, as a means to interoperate with standard Access Routers, it allows that any standard IP nodes take part in multicast communications without any change.

Interoperation with the access routers is performed by the Multicast Internet Gateways (MIG). A MIG is every ad hoc node that lays one hop away from the MANET access router (the fixed network, so any node may become a MIG at any time). The only difference between a MIG and a standard MMARP node, is that MIGs are responsible for notifying the access routers about the group memberships that are requested inside the ad-hoc network. The communications held between a MIG and the access router, are performed recurring to the same protocols and in the same way than in infrastructure networks. This provides independence of the MANET from the multicasts protocols running in the core network, hiding at the same time, the MANET operation. MMARP also allows that a MANET node can be a multicast source, distributing multicast traffic in the MANET.

MMARP operates by creating a distribution mesh similar to the one created in ODMRP. The mesh is created using a hybrid approach: multicast routes between MANET nodes are created on-demand, and multicast routes towards the multicast sources in the fixed network are created proactively.

As any reactive protocol, the reactive part of MMARP relies on request and reply phase. The request phase is performed when a node has multicast traffic to send. In this situation it issues a *MMARP_SOURCE* message that is flooded to the network in a controlled way (with the inclusion of a sequence number). This message will serve to request the network if there is interest in the multicast traffic available. When a mobile node has interest in the multicast traffic just announced, it needs to join the session. It issues a *MMARP_JOIN* message. This message is propagated hop-by-hop towards the multicast source. With this process, a shortest multicast path is constructed between the source and the destination. If only one multicast source is available, the resulting multicast routes will resemble a tree; however, if multiple multicast sources exits for the same destination, the created routes will form a mesh.

The proactive part of the MMARP protocol has the objective of advertising the MIG as a default route for any multicast source not in the ad hoc network. MIG periodically sends a *MMARP_DFL_ROUTE* towards the MANET, that is propagated just like a MMARP_SOURCE message. This message informs the nodes about the path towards the multicast sources in the fixed network. MIGs are also responsible for translating MMARP_JOIN messages into IGMP reports, that are send towards the fixed network multicast router, so that multicast traffic can enter the ad hoc network.

2.2.2 Address Auto-configuration

In order to effectively communicate in a given network, nodes must have valid and unique identifiers for that network. In an IP network, the identifiers are IP addresses, and usually have a common prefix they belong to, constituting the network. Only when a node has one, or more, address configure in its ad hoc interface, is that the routing protocols, and other IP dependent services, function properly.

Auto-configuration issues are out of scope of routing protocols. However, they depend on the existence of mechanisms responsible for dissemination of network information and configuration of nodes. The IETF MANET Working Group created the Autoconf Charter [62]. Autoconf WG soon started developing efforts to evaluate the requirements to address auto-configuration in MANET. Work inside Autoconf WG is still recent and no RFC was yet proposed.

The IP addresses inside the ad hoc networks does not need to be topologically correct [12]. Most ad hoc routing protocols do not perform routing based on an aggregate of networks, but just based on the complete IP address. However, for the MANET nodes to be able to participate in the infrastructure network services, they need to have a topologically correct (according with the fixed network that provides access). Auto-configuration protocols are useful exactly to provide IP addresses to the nodes, that are topologically correct and globally routable in the internet. Apart from the address configuration itself, auto-configuration protocols can deliver various information about the network, such as dedicated servers that provide specific services.

Although the infrastructure network already supports functionalities the configuration of nodes addresses such as DHCPv6 [28], a node entering the ad-hoc network usually has several nodes around, and probably several independent networks to use; it needs to choose one of them (either by traffic or cost considerations). Auto-configuration protocol may provide sufficient information about the network to facilitate this decision process. Moreover, DHCP protocol does not support multihop networks, like ad hoc networks commonly are.

There are several proposals that present some of the possible methods used to disseminate network configuration in MANETs. Perkins [51] proposes a simple mechanism for auto-configuration where nodes simply choose a random address and perform duplicate address detection based on a given network prefix. Jeong [39] propose a solution that differs from the previous by specifying mechanisms more suited to AODV, both for IPv4 and IPv6; it supports the existence and mergers of different network partitions. Laouiti [24] describes an

auto-configuration mechanism for isolated networks with OLSR. Wakikawa [55] proposes a method to propagate the network prefix inside the network by means of an Internet Gateway Discovery process similar to the router discovery process of IPv6, and includes the integration of MANET routing protocols with Mobile IPv6.

Jelger [38] proposes a method where the gateway providing connectivity to the Internet periodically broadcasts a message (GW_INFO), which is then forwarded by all nodes in the ad-hoc network. As the message is forwarded through the network, a hop count field present on the message is incremented. This field will be used by nodes to choose the neighbor which is closer to the gateway. This neighbor is denominated *upstream neighbor*. The Jelger auto-configuration protocol (GWINFO) flooding mechanism will create a directed tree, rooted at each gateway. GWINFO has native support for multiple gateways in the same ad-hoc, and the ability to choose one of them based on specific metrics, such as the number of hops to the infrastructure.

Secure operation of these protocols is very important in commercial environments, especially when dealing with self-configuration solutions. This prevents the advertisement of any node as a gateway, disrupting the network or increasing the chances of an eavesdropping or black hole attack. Jelger proposal has been further extended in [5], adding support for security and integration with handover mechanisms. The information on GW_INFO messages are signed by the operator and nodes are able to verify this signature using the public key infrastructure.

2.2.3 Quality of Service

Networks are build and maintained to provide services. A service can be something simple, such as e-mail and file transfer, or can be something more complex such as audio-video streaming. In todays environments, several customers use and access various services every day. Normal Internet and access networks are best effort networks. Best effort networks try its best to deliver the information to its destination; however, losses and delays can still happen. Also, best effort networks will not differentiate between high priority traffic and normal, background traffic.

The concept of Quality of Service (QoS) in IP networks provide a series of mechanisms that enable the network to provide delivery guaranties for a specified traffic, as well as differentiation between different priority flows. In a QoS enabled network, multimedia traffic, such as VoIP or IPTV, can have delay, jitter and deliver guaranties that provide the

best service possible. QoS can also help in the separation of traffic between customers with higher contracts from the ones with basic contracts, giving the former priority over the later.

When integrating an ad-hoc network with an existing commercial network, operators expect to apply the same QoS levels to users of ad hoc networks. Traditional hotspots can perform this easily by a set of rules at the access point. However, since the ad-hoc stub is a distributed and unstable environment, QoS has to be sustained in a distributed manner inside the ad hoc network.

Several protocols have already been proposed to support the delivery of adaptive services in mobile ad hoc networks. INSIGNA [16], one of the best known, uses a soft state resource management mechanism to enhance network usage. Packets transport an extra field for QoS information, which is used as an in-band signaling. The protocol supports Best Effort services and services requiring reservation with per-flow QoS support. QOLSR [1][25] is a QoS routing protocol defined to enhance OLSR. Each node gathers information related to QoS parameters such as available bandwidth, delay, jitter or loss probability. These parameters are reported to OLSR, based upon which, the MPRs create or change routes. However QOLSR is not able to limit the traffic in the network.

SWAN [9] is a QoS model that provided stateless QoS differentiation to ad hoc networks. The complete QoS solution provided by SWAN is a junction of several different mechanisms. These mechanisms are: Admission Control; Rate Control; Traffic Classification and Shaping; and MAC feedback.

The admission control mechanisms work at the source node. This mechanism controls wether the UDP traffic is allowed to enter the network or if it is blocked. This decision is dependent of the feedback given by the other mechanisms of the model. The active proving is the mechanism that provided the nodes a way to calculate the available bandwidth of a certain path. It work by sending proves from the origin to the the next hop. This packet if forwarded hop by hop to the destination. Each hop of the path evaluates the available bandwidth it has and changes the available bandwidth of the probe to the minimum acceptable. The destination node will then send a probe reply, directly to the source node, with the amount of available bandwidth on the network. The rate control mechanism is used to shape the best effort traffic class. This shaping is performed in a distributed manner in each node. The algorithm used to adjust the shapers is the Additive Increase, Multiplicative Decrease (AIMD). with this algorithm, best effort traffic will see its rate gradually increased up to the maximum bandwidth left by the real time class. MAC feedback is used to monitor the surrounding medium in order to determine the existence of congestion, and also to calculate the available

bandwidth. If the congestion is detected, MAC feedback will also mar the packets that is forwarding. When this packets arrive at the destination, a regulation process will take place to re-adjust the available bandwidth in the selected path. The classification of packets is performed at the source node. Packets are marked using the DSCP field (in IPv6). By default SWAN only supports two traffic classes, best effort and real time. In [8] an extension of SWAN was proposed to make it interoperable with the infrastructure and to support four classes of traffic.

2.2.4 Charging and Rewarding

Operators must be able to obtain profit from the development of the network and services. Profit is only obtained by charging the user for the usage of the network and their underlaying services. Charging in ad hoc networks makes most sense when the ad hoc network is interconnected with an infrastructure network. However, even when a MANET is isolated, and stand alone, there may by a need for charging users, for example, when some user has to offer to the network a distinct service. Infrastructure networks are driven by operator business models, in consequence, it is mandatory to provide support for charging the users in the ad hoc network.

The multi-hop and distributed nature (and dynamics) of ad-hoc networks requires the existence of distributed trust mechanisms, able to provide adequate information for charging and traffic authorization. Most important, these mechanisms need to be compatible and integrated with existing network authorization and charging architectures. Furthermore, ad-hoc networks also require incentives for users to participate in the forwarding process, otherwise, nodes may not forward others traffic without any benefit. Such incentives can be provided in many forms, like, for example, credit or service discounts.

The solutions proposed by the research community envision scenarios where ad-hoc networks are integrated with an infrastructure supporting authentication, authorization and charging mechanisms. Salem [4] envisions ad-hoc extended cellular networks, where base stations are capable of charging, rewarding and enforcing profile policies on packets generated. In order to achieve this level of control, it proposes all traffic to cross the base station, independently of its origin and destination. SCP [14],[13],[11] proposes the creation of a distributed mechanism, actively marking packets with a proof that is updated at each forwarding node and then reported to the network operator, with intrinsic class differentiation. The proofs are built and updated using a defined set of rules and supported by cryptographic

signing and verification primitives. SPRITE [23] assumes that nodes have enough storage capacity to store traffic proofs. These proofs are later traded at a bank for credit when the node is connected to a high bandwidth medium. PACP [2] improves many of SCP deficiencies (overhead, variable packet size) by encoding the route in a polynomial included in the packets, and securely updated at every node. Upon reception of the charging information on the infrastructure network, the appropriate charging and rewarding actions may be applied. These actions can take in consideration many individual parameters, like individual user profile, service description, QoS parameters, route length, time frame or data amount. Also, PACP supports distributed access control, allowing the operator to control which flows are allowed between each nodes, without sacrificing routing.

2.3 Ad-hoc networks as hotspot extensions

Enabling a mobile ad-hoc network to be a hotspot extension and integrating it into a 4G scenario [18], [20], [19], implies interconnecting it with the infrastructure network and supporting basic its mechanisms. This interconnection will provide the MANET to be a valid extension of the overall operator architecture.

The previous sections presented and discussed the MANET special characteristics (2.1) and its special requirements (2.2). From those discussions its was made clear what are the functionalities that need to be available in the mobile ad hoc network so that it can provide a good set of services.

For a MANET to participate as stub network in a larger network infrastructure, is has to meet some of the requirements of the larger network it integrates to. Figure 2.1 shows the core network functionalities that the MANET needs to support.

Figure 2.1 shows the basin layout of the network. The network is divided in the Core Network and the various Access Networks. This two networks form and Administrative Domain, that can be compared to an operators network. In the core network are located the servers that control and aid all the operations performed by the mobile nodes operations inside the administrative domain. In the Access Networks are located the mobile nodes, and is through this networks that the mobile nodes (and its users) access the subscribed services.

The Authorization, Authentication, Accounting, Audit and Charging (A4C) server controls the admission of nodes in the network, and monitors the network operation, so that the network can enforce the contract that the mobile node user has purchased. This server is

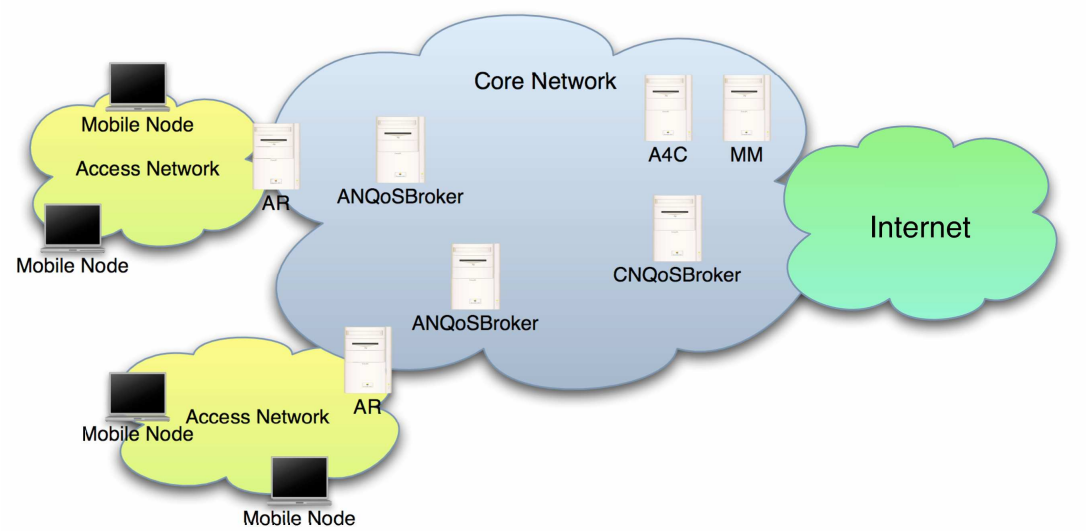

Figure 2.1: Core network and access network diagram

contacted each time a node joins the network, and each time a node stars using any of the network offered services. The Core Network QoS Broker (CNQoSBroker) is the server responsible for controlling the resources of the core network, for managing the inter-domain signaling messages and for the multicast support. The Mobility Manager is the server responsible for managing mobility inside the administrative domain, and for aiding the mobile nodes to move to other administrative domains. Finally, the Access Network QoS Broker is responsible for managing the resources of the access network. It also helps in the mobility management, in the admission control and in the in the admission control (serving as a proxy to A4C).

The MANET needs to interoperate with all of these servers in the core network, and support their functinalities, so it can properly operate in the extended hotspot scenario as a stub network.

2.3.1 Scenarios for hotspot networks

Ad hoc networks as hotspot extensions are not supposed to be used as a primary mean for accessing normal networks services. As shown in chapter 3, the performance of a multi-function and multi-service mobile ad hoc network does not permit the usage of the services that nowadays have most demand on the internet, namely heavy multimedia services and alike, that tend to be very bandwidth hungry. However, most traditional internet services such as light web browsing, basic email services and Instant Messaging are better suited for

low bandwidth networks.

One key factor of every hotspot network is its coverage area. Although the number of wireless access points is increasing rapidly, with more developed cities having a large area of coverage that provides internet access, coverage area tend to be scarce. In this scenario, ad hoc networks have an window of opportunity. The MANET multihop nature has the ability to extend and spread the network for large geographic areas, requiring less points of interconnection with the core network. Moreover, not all of the devices in a MANET will not have significant mobility (inside the network), which will permit the network to have more stable routes towards the gateways.

Typical deployment scenario for a ad hoc network as hotspot extension can be as a complement for traditional wifi hotspots, in where the ad hoc will be used to give network access to nodes that move outside the coverage are of WiFi access point (AP), either because of a node's movement to the far side of the AP coverage, or in the transition from one AP to another. In this scenario, the nodes and the network have to support the movement of nodes from a wifi network (infrastructure) to a ad hoc network, and vice versa, without loosing connectivity or its current open sessions.

Other type of scenarios that justify the usage of mobile ad hoc networks as hotspot extension can be the in a road. MANET can be used to provide access to the vehicles, and its passengers, to the core network services. Due to the large geographical area that a road (highways for example) has to cover, it is more easily to deploy several, far away from each other, MANET gateways, than a set of WiFi access points.

2.3.2 Auto-configuration

When interconnected with an infrastructure network, a MANET needs to provide a mechanism that allows nodes outside the MANET to reach the nodes belonging to the MANET. This mechanism implies the usage of nodes identifiers that can be valid outside the MNAET. IP networks rely on the IP address to both, locate the nodes and identify them, so the MANET mobile node identifies have to bee IPv6 addresses of global scope. Auto-configuration protocols provide a mean for nodes to obtain IP addresses that are globally reachable, and topologically correct in relation the the Internet.

In the extended hotspot scenario, the auto-configuration protocol has to be configured by, or communicate with, the core network, so it can distribute and announce prefixes that are topologically correct with the outside of the MANET. Auto-configuration protocols have

to operate much like the Dynamic Hosts Configuration Protocol in traditional IP networks. After a node is configured with an valid and globally routable IP address, it is able to establish sessions with nodes outside the MANET.

2.3.3 Unicast and Multicast Routing

When two MANET nodes have a on-going communication, the routing protocols that operate in the MANET permit that they can communicate between then. Traffic is forwarded between communication peers, with the help of the other nodes in the MANET. Moreover, routing protocols permit that routes towards any node in the network can be found (as long as there is a path that lead to that node).

When operating in a hotspot scenario, the routing protocol in use must be able to differentiate which nodes are inside the MANE and which nodes are outside of the MANET. This distinction can be made by looking to the target node address prefix. When a route is requested to a node that is located outside the MANET, the routing protocol will forward the request to the gateway. Gateways have to operate as routers between the two networks, and convert MANET route requests into traditional infrastructure networks route discovery mechanisms. In the same way, the gateways must be capable of translating the infrastructure network's route discovery mechanisms into MANET mechanisms, so that outside-MANET nodes can communicate with nodes inside the MANET.

Multicast routing must support the same requirements made to unicast routing. The multicast routing protocol in use must by able to identity multicast sessions that are been generated outside the MANET. The multicast session join request for those sessions must be forwarded to the MANET gateway. The gateway must interoperate with the multicast router in the core network, so that the multicast traffic can be directed to the MANET.

2.3.4 Quality of Service

Albeit, quality of service (QoS) can be provided only inside the MANET, its is most useful when it enables interoperation with the infrastructure network in a hotspot scenario. This way, MANET nodes can perform reservations for acquire service with more quality and guaranties.

The MANET gateway will have to translate QoS reservations requests coming from the MANET to reservations in the core network. In order to do that, the gateway will have to

interoperate with the QoS infrastructure present in the core network (ANQoSBroker). When reservations are performed, the gateway will have to enforce them both, in the MANET side as well as in the infrastructure network.

2.4 Summary

This chapter has introduced the ad hoc network, and the Mobile Ad hoc Networks in particular. The specific characteristics of ad hoc networks were presented, and based on them, the special requirements and the protocols that fulfill them were presented and described. Namely, solution were presented for auto-configuration, for routing, both unicast and multicast and for quality of service. The necessary extension needed so that MANET can participate in a hotspot scenario as a stub network were also presented, along with the changes that need to be done to interoperate with the core network servers.

Chapter 3

Experimental Evaluation

The previous chapter described the ad hoc network concept and the most common protocols used to provide to it a variety of functionalities. In section 2.3 the necessary extensions that the protocols need to support in order to make the ad hoc network function like and hotspot extension were described. The primary objective of this chapter is to present the experimental evaluation of an ad hoc network used as a hotspot extension. In order to perform the experimental evaluation, a testbed was built. This testbed is constituted with different personal computers and has all the main features required for a complete hostspot extension. Auto-configuration, unicast routing, multicast routing, quality of service and charging and rewarding support are the services that are part of the testbed and which were evaluated. The results of the evaluation are also presented, as well as an analysis of the obtained results.

This chapter is organized as follows: section 3.1 describes the Mobile Node architecture, describing also the services offered and correspondent protocols that provide each service. Section 3.2 describes the ad hoc gateway architecture in a similar manner to the mobile node's architecture. The software environment in were the chosen protocols where developed and tested is described in section 3.3. In section 3.4 the testbed description is presented, in this section the hardware used for the tests and the selected network topologies are presented and explained. Section 3.5 presents the obtained results during the evaluation, presenting also the methodology developed for performing the tests. The discussion of the obtained results from a point of view of a the full service network is presented in chapter 3.6. Finally, in section 3.7 a conclusion for this chapter is elaborated.

The developed and tested testbed is part of the network demonstrator of the IST Project DAIDALOS [60].

3.1 Mobile Node Architecture

The main functionalities that need to be present in the mobile node, in order to benefit from the operators hotspot are: auto-configuration, unicast and multicast routing, quality of service and charging. With these functionalities present, the MN can participate in the operator hotspot, while using services such as real-time voice and video, mixed with the rest of the network's bulk traffic.

The auto-configuration mechanism is required in order to enable nodes to discover hotspots and auto-configure their addresses correctly. After successful configuration, unicast and multicast routing is used to provided basic network functionality. The mentioned multimedia services (voice and video traffic), require that their traffic is differentiated from the network bulk traffic; in order to do that, the network needs quality of service support. Finally, operators must be able to account and record each user's network usage, in order to apply contracted profiles agreed with each user. The QoS and Charging mechanisms are performed in a distributed manner, but without disclosing the user profile, so that traffic must not be forced to cross the gateway.

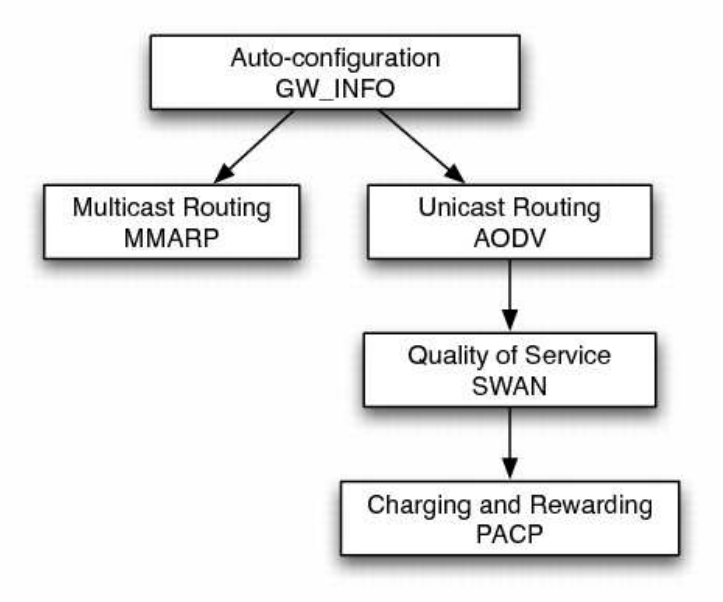

Figure 3.1: Functional architecture for the mobile node

Figure 3.1 shows the functional architecture of the mobile node, showing for each functionality, the chosen solution. The next subsections will better detail each protocol.

The general architecture for the mobile node and the gateway is show in figure 3.2.

This figure represents the principal elements of the architecture and their interaction with infrastructure elements presented in the left side of the picture.

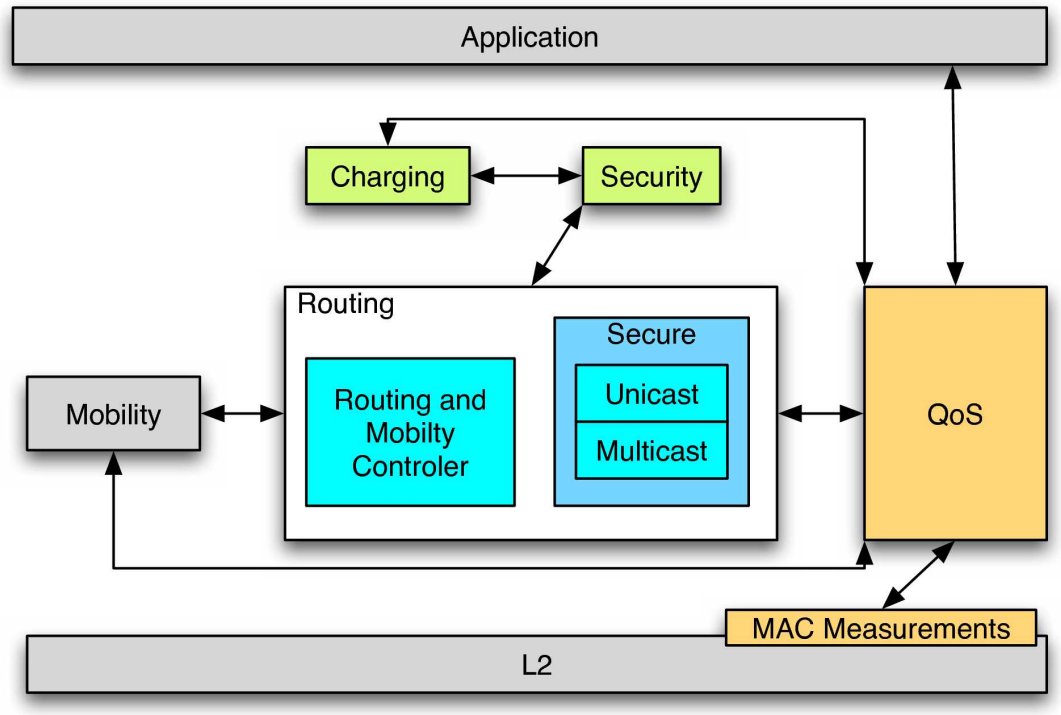

Figure 3.2: Mobile Node general architecture

3.1.1 Auto-configuration and Gateway Awareness - GW_INFO

The chosen solution to provide auto-configuration is the proposal presented in [37] and later extended in [5]. This was the most appropriate solution to the deployment environment, as the others lacked in security [56] - [46], dependence of the routing protocol [51] - [46], or adequacy to hybrid scenarios [51] - [46].

Whit the protocol described in [5], named GW_INFO (for GateWay INFOrmation) the nodes are able to choose which gateway to use, based on different criteria, not just the distance, and also change between gateways, provided that they support Mobile IPv6 [41] in order to maintain global connectivity.

3.1.2 Unicast Routing - AODV

The chosen protocol to perform unicast routing is the AODV protocol [50]. The reasons for this choice are based on the type of scenario envisioned (see section 2.3.1) and the availability of the implementations at the time the decision was taken. This will guarantee a better service and less integration problems. The AODV implementation from Upsala University, AODV-UU [73], in its 0.50 version, is the chosen implementation. Some changes were performed to the *vanilla* version in order to support the address auto-configuration and dynamic change of IPv6 address. Moreover, some modules, such as the QoS modules and the charging modules, needed to get information about the routing tables, more specifically, the next hop node for a given route. Albeit this kind of information can be retrieved directly from the operative system routing tables, AODV is able to provide a set of more useful information, such as alternative routes. One more change was made in order for AODV to wait for authorization modules to verify that a given route is valid (authorized).

All the changes made did not alter the basic algorithm of the routing protocol, so no performance penalties are expected, especially in low mobility scenarios.

3.1.3 Multicast Routing - MMARP

After evaluating the various proposals for a multicast routing protocol, only one can offer multicast traffic to the ad hoc network maintaining compatibility with the rest of the internet, which typically run IGMP [29] and/or MLD [54]. That protocol is MMARP [17], which allows to provision to the ad hoc cloud the same multicast services provided to infrastructure nodes without any changes to the existing architecture and with the addition of security mechanism [10].

This transparent functionality is obtained with the creation of Multicast Internet Gateways (MIG), which are ad hoc nodes directly connected to the gateway. These nodes are responsible for translating MMARP signaling into IGMPv6 signaling. The MIG performs a dual role, it communicates with the gateway in order to communicate the interest revealed by other MMARP enabled nodes at the same time that announces itself to the ad hoc nodes as the default multicast gateway, and informing about the path towards multicast sources in the fixed network.

Albeit special in their operation, a MIG is not a dedicated node, any ad hoc node MMARP enabled can become a MIG at any time, and does so when directly connected to

the gateway. Apart from this proactive behavior, MMARP also has a reactive component to create and maintain the distribution tree over the ad hoc network, using *Join* messages towards the source to create a multicast shortest path.

3.1.4 Quality of Service - SWAN

Among the multiple well-known protocols to deliver QoS in ad hoc networks, SWAN [9] proves to be one of the best choices [22]: it has lower overhead than INSIGNIA[16] and is the QoS protocol that performs better with AODV. In order to allow QoS inter-operations between the ad hoc and infrastructure networks, the base SWAN signaling was adapted and extended [8]. These extensions provide SWAN with QoS signaling based admission control and support for multipath probing. The differentiation model was also extended to support several service classes as well as congestion feedback between then. The extended differentiation model considers four different traffic classes: critical real-time traffic, less demanding real-time traffic, non-real time traffic and regular best-effort traffic. Each of these classes will have assigned a certain amount of bandwidth, except of course the best-effort, that uses the remaining bandwidth.

In Figure 3.3 is depicted the differentiation model composed by a classifier and by a cascade of priority schedulers, shapers and queues associated to each traffic class. The delays are applied to each packet through a leaky bucket shaper, whose rate is controlled by an AIMD algorithm having the lower level classes delay as feedback.

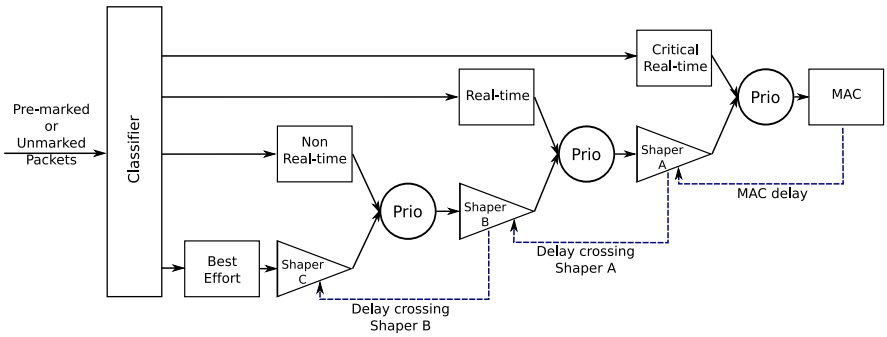

Figure 3.3: SWAN differentiation model

The implementation used follows this extended model supporting four traffic classes. This software also provides extended session admission and integration with external authentication and authorization servers.

3.1.5 Charging and Rewarding - PACP

PACP [2] revels itself as the best choice for a charging protocol, given the integration requirements of the scenario. PACP is also the best choice when no ideal rewarding is required, that is, when rewarding for 100% of the traffic is not the main concern. The other considered proposals fail to comply with some of the main requirements of the used scenario: not providing proper integration with the infrastructure network [23], large overhead, as the number of nodes increase [15] or even the usage of non-optimal routes [4]. The features PACP provides are: correct charging and rewarding information, secure process of proof creation and delivery, usage of optimal routes, small network overhead and small processing requirements in all modes. This in conjunction with the ability to interoperate with the infrastructure networks makes PACP the best choice.

The operation of PACP is as follows; the identification of the route followed by a data packet is implicitly included in a fixed size field inside each data packet. This identification will be updated in each node in the ad hoc network, as the packet flows to the destination. These fields are cryptographically secure (to prevent attacks from malicious nodes), so when this information is corrupted the packet will be dropped (by the node forwarding it). The last node in the data flow's path, just after the destination, denoted "last forwarding node", is the node responsible for sending the proofs to the gateway. These proofs contain information about the packet travelled path. The gateway collects the proofs and sends then to the authentication and accounting server, so that the truthfulness of the information can be verified, recurring to the cryptographic information contained in the proofs. With the proofs information the authorization and accounting server knows what nodes have sent and received traffic, and what nodes have forwarded a specific packet, these nodes will then receive rewards for the forwarding performed.

PACP in conjunction with proper gateway control processes can provide the tools required to check the behavior of nodes inside the ad hoc network, eventually leading to creditation/reputation schemes, developed with the aid of the network operator.

3.2 Gateway Architecture

The architecture of the ad hoc gateway (GW) is very similar to the one of the mobile node (MN), since it is the gateway that provides most of the services that the mobile node will access. The gateway is an element belonging to the operators networks, so it is up to the

operator the proper configurations of the gateway and of the services that it offers. Some services can be suppressed completely (multicast for example) or be offered only to some extent (reduced QoS traffic classes). In the presented testbed however, all the services will be available, and fully operational.

In terms of software modules, much is shared between the MN and the GW, changing only the behavior of the modules and existing only a few exceptions. The auto-configuration and unicast routing mechanisms are present so that the ad hoc network can be created; the QoS modules will be present in order to establish the bridge between the ad hoc and the infrastructure networks; charging and multicast modules will also be present, in order to provide the desired functionality.

The gateway general architecture is presented in figure 3.4. This figure is similar to the one presented for the mobile node general architecture. The ad hoc elements and their interaction with the infrastructure specific elements are also show. The infrastructure elements are on the left of the image.

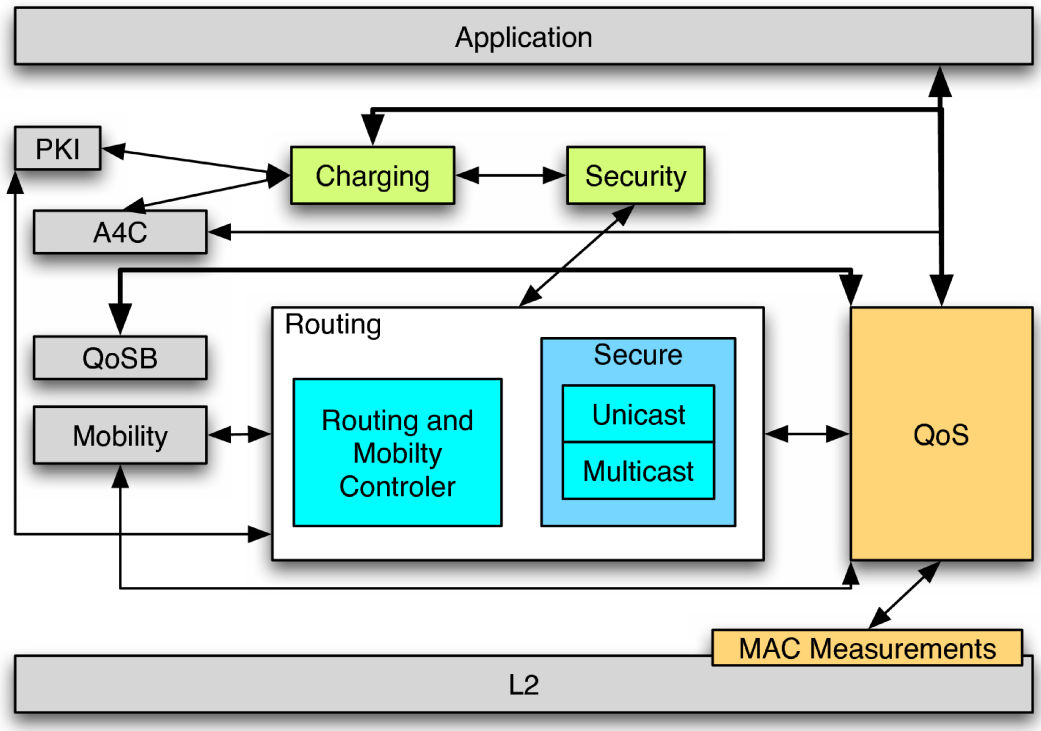

Figure 3.4: Gateway general architecture

It should be noticed that no choices needed to be done regarding the software or protocols running in the gateway; the same choices for the MN apply in the gateway. Instead

the support for a gateway was also one of the requirements considered when choosing final solutions.

3.2.1 Auto-configuration - GW_INFO

The auto-configuration function in the gateway has only the task of periodically send the GW_INFO messages announcing the gateway. Each message contains the IPv6 address of the gateway, the network prefix and the network mask that will be used by the MN's to configure their IPv6 address.

Additional functionalities of the auto-configuration module in the gateway are only related to the type of algorithm used to distribute the messages (proactive and reactive) and with the type of method used to solve ties when multiple upstream neighbors were present (distance based and prefix based tie solving algorithms). However, any change made in the parameters of the auto-configuration algorithm need to be propagated to the mobile nodes manually, since the available implementation does not have dynamic parameters configuration via network messages.

3.2.2 Unicast Routing - AODV

The AODV module in the MN is responsible for finding routes to/from other nodes in the network, and also perform the forward of the actual packets. In the GW it is responsible for forwarding the packets between the infrastructure network and the ad hoc network, acting more like an Access Router. The AODV software module will listen for route request messages (RREQ) coming from the ad hoc network and analyze the destination address. In case the address does not belong to the addresses present in the ad hoc network, it then triggers a search in the infrastructure network, recurring to the standard IPv6 mechanism. In a similar way, when the infrastructure network is looking for nodes whose address belongs to the address space of the ad hoc network, then it needs to translate them into a route discovery process inside the ad hoc network, in order to find the node.

Equally to the MN, the gateway version of AODV also has interfaces with the modules supporting the other functionalities, in order to offer the information needed by them.

3.2.3 Multicast Routing - MRD6

The multicast function of the gateway is not performed by the MMARP protocol, not because of the usage of a different protocol, but because the MMARP protocol was designed to not run on the ad hoc gateway, but relying is the gateway's natural ability to route packets. Since the target platform (see section 3.4) does not natively support routing of multicast traffic, the Multicast Routing Daemon (MRD6) [70] software was used.

MRD6 provides the desired interoperability with the infrastructure network, listening to MLD[54] protocol on the ad hoc network, and then translating to the protocol used in the infrastructure network. The ad hoc gateway is then a different entity from the multicast internet gateway (MIG). The MIG is the last node in the network, before the gateway, that is running MMARP. The MIG listens to MMARP protocol from the ad hoc network and then sends MLD protocol messages towards the GW, in where MRD6 in turn will *pull* the multicast traffic to the ad hoc network.

3.2.4 Quality of Service - SWAN

The QoS modules present in the gateway are very similar to the ones in the mobile node (see section 3.1.4) both in their architecture and in the functionality. The most important difference is that, being the gateway also the access router for the network, it will not generate traffic, only will forward traffic; consequently, the QoS client present in the mobile node (responsible for interacting with the applications to obtain QoS requirements) is replaced in the gateway by the QoS Manager.

The QoS Manager will interact with the Access Router Manager Software (ARM) and with the QoS Broker in the infrastructure in the same way all other Access Router's QoS Manager do, enabling that the reservations request coming from ad hoc nodes can be translated and propagated in the infrastructure.

3.2.5 Charging and Rewarding - PACP

Regarding PACP implementation [58], the MN and the GW are not very different in terms of the running software, most of the code is shared among them. The main difference is the main protocol logic. Mobile Node logic is implemented by the Charging Agent and the Gateway code (called Access Router in the PACP context) is implemented by the Internetwork

Adapter. The main functionality of these modules is described in section 2.2.4

3.3 Software Environment

All software used was developed and/or modified under a *GNU/Linux* environment. The official distribution of the Testbed was Mandrake 10.1 Official [67] (now discontinued), but recurring to a vanilla 2.6.8.1 Linux kernel, enhanced with modifications required by some of the testbed modules. There are several patches applied to the vanilla kernel, namely:

- Support for DSCP marking using Netfilter [71]: Used by the QoS modules to mark all packets for posterior identification and differentiation.
- Support for Token Bucket Queue.
- Host AP wireless driver [65]: the driver of the wireless chipset used.
- Netlink Multiplexer.
- A IP6_QUEUE multiplexer: Standard kernels only support one client (local client) at a time in the IP6_QUEUE netfilter table. This patch provides the support for multiple clients simultaneously.
- The MACKILL [64] network topology emulator (custom version).

With exception of (parts of) the AODV-UU and Host AP driver, all other patches were developed and tested inside the Daidalos project [60]. Figure 3.5 shows a partial vision of the software modules used, mostly focusing on the customized functions described above, and in the functional modules described in the earlier sections.

Apart from the already mentioned software, several custom scripts were also developed, in order to easy the management of the overall software. The integrated system proven difficult to manage, following the development phase; the integration of all the software was a demanding task. The level of integration achieved was sufficient for running small, short-duration, trials.

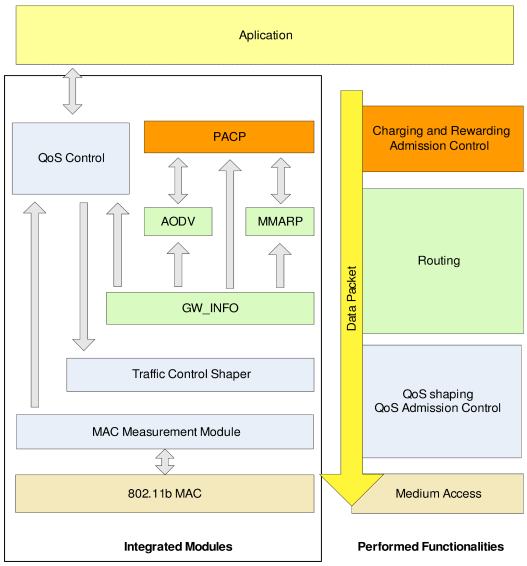

Figure 3.5: Software architecture

3.4 Testbed Description

3.4.1 Hardware description

The testbed is composed with various types of PC's ranging from the most simpler laptop with a 1.2 GHz CPU and 256MB of RAM to the more powerful Edge Router with an AMD Athlon^{TM} 64 Processor 3000+ (1.8MHz) with 1GB of RAM. All PC's have plenty of storage space.

The lower limit in terms of hardware requirements was established by the ad hoc nodes, since using PC's with (even) lower capabilities will cause too much slowness in the system altering the results. These specifications do not reflect typical, resource limited, (current) ad hoc nodes, but are only suited to the extensive testing possible in a lab, or to yet-to-be-developed small form factor PDAs.

All terminal (MN and GW) machines are equipped with two network interfaces: one wireless and one wired. The wired interface is used to provide remote access during the tests and for administrative tasks. The communications performed for the test are restrained to the wireless interface in ad hoc mode.

The wireless interfaces used were Prism2.5 802.11b cards with the following configuration parameters: channel 12, rate fixed to 11Mbits and RTS/CTS threshold of 1 byte.

The bit-rate limitation was used to increase reliability, avoiding bit-rate changes and support a channel with bit-rates easily handled by the mobile nodes. Channel 12 was selected for interference minimization. Regarding the ad hoc network topology, and since the tests were performed in the lab, the MACKILL [64] was used to create the desired (and emulated) topology (as mentioned is section3.3).

3.4.2 Network Topologies

Since the evaluation of the testbed is focused on the performance, the network topologies used were chosen in order to evaluate the influence that the number of nodes has. Due to our limited number of computers, the chosen topology is a *string topology* where the nodes are connected sequentially to only two neighbors, thus maximizing the number of hops.

Figure 3.6 shows a representation of the topology used, where six machines show the linear topology used. Node1 is the gateway of the Ad hoc network and Node6 is the last hop. Using this topology, without interfering traffic, we can show a bound to the maximum achievable performance of the network.

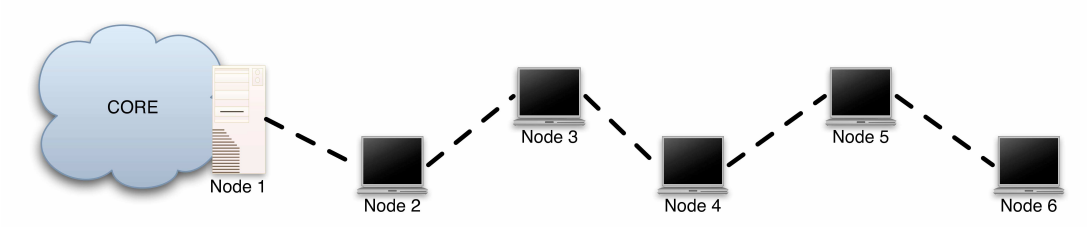

Figure 3.6: String topology used in the tests

This topology was used in two different locations, indoors and outdoors. The indoor topology was deployed in one of the rooms of the IT-Aveiro [66] building, which is a roughly square building with around 1300 m^2, and normal office/lab divisions. Since there is not enough physical space inside the building to create the desired topology without the nodes interfering with each other, the MACKILL [64] tool was used. MACKILL filters the packets, in kernel, based on the source MAC address. With the right configuration, MACKILL provides the desired string topology, at a logical level (radio interference still exist however). A final note to refer that all the nearby Access Points where turned off when the test were performed.

The outdoor topology is represented in Figure 3.7; as can be seen, the nodes are spread across a wide space, creating the desired topology naturally, without the aid of special soft-

ware. With this topology the objective was to evaluate the influence of the radio interference present in the indoor test and also the stability of the ad hoc network when used in more real-life network arrangements.

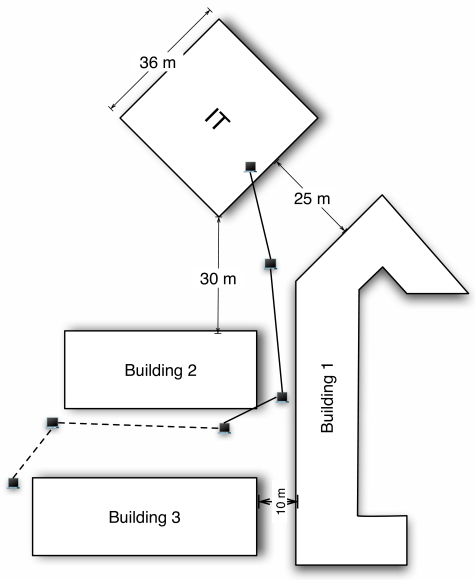

Figure 3.7: Outdoor topology

3.5 Experimental Results

This section will present the results of the performed tests. The aim is to measure delay, jitter and overhead, for a specific set o traffic profiles, specially targeting multimedia applications. Three UDP traffic profiles were defined, according to different bit rates, 64kbps, 128kps and 256kbps, in order to evaluate the network without being in stressful situations. These traffic profiles model close enough voice and video applications.

The experiments were designed and performed with the intention of measuring the incremental addition of nodes to the network and also the incremental addition of functionalities to the nodes. The experiences are divided in various groups, and each group is divided in various steps.

The group division is performed according to the functionalities to be evaluated. In the first group only the auto-configuration modules are running in the nodes; in the second group, the auto-configuration module is added to the unicast routing module; In the third group the QoS functionality is added, hence three functionalities are evaluated simultaneously; to the

functionalities present in the third group, the charging and rewarding functionality is added, constituting the fourth group. Finally, the fifth group contains only the auto-configuration and multicast routing modules.

Inside each group, various steps are performed. What defines a step is the number of nodes that are participating in the experience. Hence the first step is always considering two nodes: the gateway (Node1 in figure 3.6) and the first mobile node (Node2), with traffic flowing from the node to the gateway. For the next step another node is added, and so forth until the last step which involves all the nodes with traffic also flowing from the last node to the gateway. The throughput is calculated differently, since it aims to measure the maximum bit rate that can be sent in the network, without loosing packets and without taking into consideration delay or jitter. For measuring the throughput it was used a method of trial and error: starting in the one hop connection (Node2 to Node1) with a theoretical estimation for the bandwidth, the throughput is adjusted to the maximum one with no losses. For the next hop, the previous value for throughput was divided by half, and later adjusted. This procedure was repeated for all hops, achieving a final figure for the throughput in all hops. The MGEN tool [68] was used to generate the constant bit rate (CBR) traffic flows. This tool is very versatile, and has facilitated the process of automating the experiments. For each step, 5 runs were made, with 300 seconds for each run.

In Section 3.5.3 a comparison between the performance of unicast routing for both the indoor and outdoor scenarios is used to determine the difference of one scenario over the other. Based on the results obtained, which have no main differences between each other, it was decided that the remaining results were obtained only in the indoor scenario.

3.5.1 Auto-configuration

The functionality provided by the auto-configuration module is the configuration of the IPv6 address, so no traffic can be sent in the network, because no routing can be performed. The measured metrics were the overhead caused by the protocol in the network and the time needed for the address to be configured.

Measured overhead was 992 bps per link, which, for a 64 Kbps bit rate, represents 1.44% of the data. Since for each link, each node sends a GW_INFO message per second, and the GW_INFO messages are of constant size, the overhead of the auto-configuration module tends to be constant, only depending on the number of nodes present in the link.

Auto-configuration time takes an average of 2 seconds and represents the time since

the reception of the first GW_INFO message and the transmission of the first GW_INFO message to other nodes, that is, when the node is fully configured. Each node sends a new message once each second so, when a node moves inside the ad hoc network, it receives a new GW_INFO message, from a potential new upstream neighbor, after 1 second, in the worst case scenario. Generally, auto-configuration was seen not to have a large impact in network performance.

3.5.2 Multicast Routing

As was said earlier, the multicast experiments were performed only with GW_INFO and MMARP running, no more software was needed. Also in the multicast experiments, the source for the flows is Node1 (see figure 3.6). In the first step Node2 sends a *Join* message to start receiving the multicast traffic, then in the second step, Node3 sends also a *Join* message. When Node2 receives this join, it automatically becomes a MIG for that network. In the remaining steps all other nodes send a join message and start receiving the traffic. Each step is repeated several times (several runs per step).

The first of the metrics evaluated is the throughput of the network. The Table 3.1 shows the variation of the throughput with the addition of nodes to the network. It is observed that the throughput in a direct link connection between two hops is 1223 Kbps, which corresponds to the effective user data transmission. The real throughput in the network is slightly higher due to additional headers and RTS/CTS mechanism. When the five hops are considered the throughput available to the last hop is diminished to 76 Kbps. Since these tests were performed in the indoor topology, the results are affected by the radio interference. It is worth notice that, by its nature, MMARP does not need special protocol messages between the gateway and the first hop (direct neighbor of the gateway), only standard IPv6/MLD mechanisms are necessary. This reduces the overhead in the network, improving also the overall throughput.

Hops	multicast (Kbps)	unicast (Kbps)
1	1223	1222
2	672	559
3	291	322
4	191	204
5	76	122

Table 3.1: Throughput: routing and auto-configuration

The delay and Jitter were the next metrics measured in the testbed. The objective of

this test is to evaluate the impact of multicast routing in the traffic for different configurations of the testbed when the network is not fully congested. Tables 3.2 and 3.3 show the obtained results. It can be seen that the performance of the multicast routing for the first hop is very similar for the tree traffic profiles used, since the available bandwidth is much larger than the one used. When the number of hops increases, delay also increases for the two lowest bit rate traffic profiles studied (64 and 128 Kbps). The third flow (256 kbps) shows large delays for hop count larger than 3, when the maximum throughput is achieved. It can also be observed that the delay value for a direct connection is smaller than the delay increase with the number of hops. For 256 Kbps flows, delay reaches values larger that 100 ms, which is not acceptable for voice; however, video streaming can still be supported.

Delay (ms)	64 Kbps	128 Kbps	256 Kbps
1 Hop	3.527	4.184	4.809
2 Hops	8.910	9.912	31.642
3 Hops	13.194	45.474	113.267
4 Hops	16.941	67.027	194.941
5 Hops	21.619	82.823	252.608

Table 3.2: Delay: multicast routing and auto-configuration

Jitter (ms)	64 Kbps	128 Kbps	256 Kbps
1 Hop	0.227	0.224	0.221
2 Hops	1.669	1.930	10.586
3 Hops	0.841	25.286	20.306
4 Hops	1.142	25.119	22.246
5 Hops	1.374	21.743	23.683

Table 3.3: Jitter: multicast routing and auto-configuration

Table 3.4 shows the packet loss results. As can be observed the losses reach values higher than 10% for communications of 256 Kbps traversing more than two hops. This tendency can also be observed with the 128 Kbps traffic profile beyond 3 hops.

A final metric measured was the overhead introduced by the MMARP and GW_INFO protocol. This overhead is 3.94% in 64 Kbps of traffic, which indicates that the overhead added by MMARP alone is almost twice the one introduced by the auto-configuration protocol.

Loss (%)	64 Kbps	128 Kbps	256 Kbps
1 Hop	0.24	0.35	0.54
2 Hops	3.13	2.39	3.40
3 Hops	2.06	8.00	11.85
4 Hops	2.38	8.04	22.89
5 Hops	2.82	11.72	33.03

Table 3.4: Packet loss: multicast routing and auto-configuration

3.5.3 Unicast Routing

Unicast routing, along with address auto-configuration, is one of the key functionalities needed in the network; without it no other functionality can be used (except from multicast routing).

The first measurements taken were related to the maximum network throughput, as a function of the number of hops present between the sender and the receiver. Table 3.1 shows these results. As expected, the throughput decreases with the numbers of hops. Further, it can be seen that for the one and two-hop counts the achieved throughput is bellow the maximum achieved by MMARP. This effect is due to the lower overhead presented by MMARP in the first hops. However, as the number of hops increases, the maximum throughput of AODV is higher that the one of MMARP, due to the increase of MMARP overhead.

The second measurements are from the packet delay (table 3.5) and jitter (table 3.6) for the different studied traffic profiles. As can be observed, both delay and jitter slightly increase with the increase of flows' bandwidth and with the number of hops. In the fifth hop, with the traffic profile of 256 Kbps, the delay introduced by AODV is higher that the one introduced by MMARP, which is due to the way the protocols work. AODV tries its best to deliver the packets to its destination, so when no more traffic can be sent to the network, it is buffered and sent later, hence introducing delay. MMARP discards the packets that cannot be delivered, increasing then the packet loss count.

Delay (ms)	64 Kbps	128 Kbps	256 Kbps
1 Hop	4.474	4.606	4.535
2 Hops	9.058	9.242	9.045
3 Hops	13.968	15.036	17.691
4 Hops	19.578	20.924	97.502
5 Hops	23.619	24.248	1333.563

Table 3.5: Delay: unicast routing and auto-configuration

The overhead introduced by AODV and auto-configuration protocols is of 2.38% per

Jitter (ms)	64 Kbps	128 Kbps	256 Kbps
1 Hop	0.560	0.741	0.697
2 Hops	1.254	1.236	0.997
3 Hops	1.248	1.434	1.835
4 Hops	2.205	1.975	13.456
5 Hops	1.452	2.228	21.474

Table 3.6: Jitter: unicast routing and auto-configuration

hop with 64 Kbps of traffic flowing in the network, which is similar to the one introduced by MMARP. The additional overhead introduced by AODV alone is of 0.94%.

Hops	Outdoor (Kbps)	Indoor (Kbps)
1 Hop	1223	1222
2 Hops	432	559
3 Hops	258	322

Table 3.7: Unicast routing and auto-configuration: indoor and outdoor throughput

Unicast routing was also evaluated in the outdoor scenario. The results were used to determine the impact of using one scenario over the other. In Table 3.7 the throughput results for the outdoor topology are presented. It can be noticed that there is no large difference between the indoor or outdoor scenario, except a throughput decrease with the increase of the number of hops. This results seems to contradict the expected behavior, since in the outdoor scenario there is no radio interference between all the nodes (only between neighbors), but in fact in the outdoor scenario the distance between nodes increases, and so do the number of transmission errors, causing also a decrease in the throughput. Similar results were also obtained for the delay and jitter values (not presented here). Based on these results, it was decided that the remaining test would be conducted only in the indoor scenario.

3.5.4 QoS

Continuing with the incremental evaluation of the modules when working together, Quality of Service (traffic control and service differentiation) is now evaluated. In terms of traffic control, the maximum achievable throughput and the influence of the number of nodes was evaluated. The maximum throughput decreases with the number of hops, from 1.2Mbps (one hop) to 120 Kbps (5 hops), similar to previous results. These values were expected, since when the traffic is marked real-time, there is no action taken by the QoS modules, and processing time is virtually inexistent.

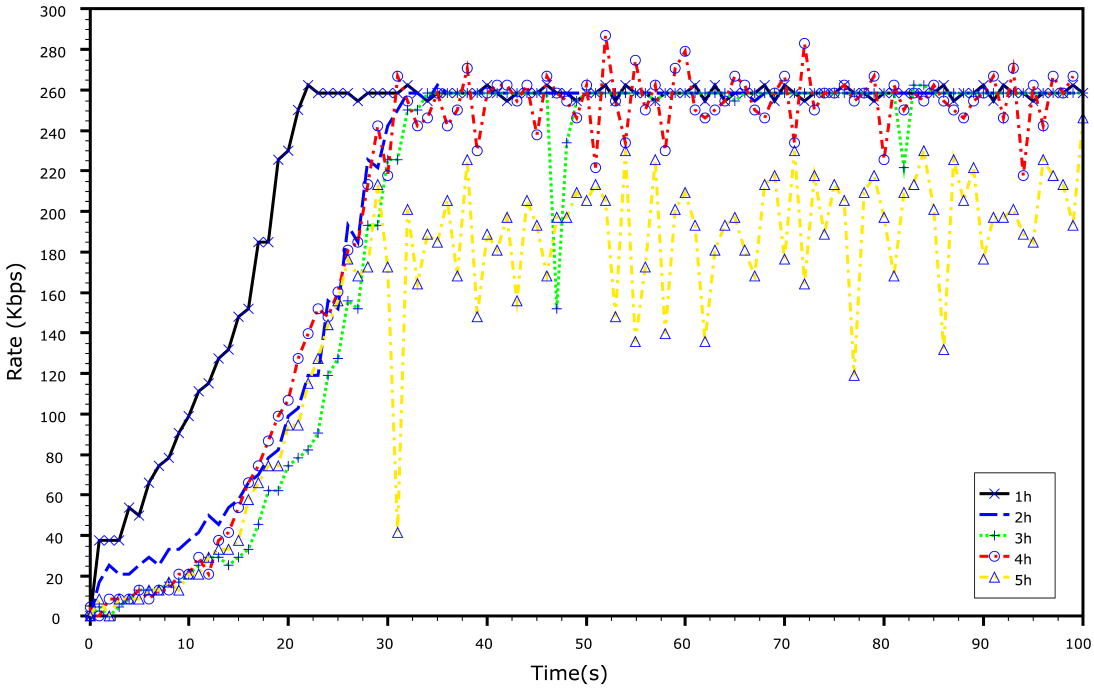

Figure 3.8: "Less demanding real-time" rate variation

Advancing further, Figure 3.8 shows the influence of the number of hops in the differentiation time for the lest demanding real-time class (class with priority just bellow the real-time class). In all cases the flow bandwidth is 256 kbps, the transmission always starts at time 0 seconds, and immediately at the maximum bit rate. First it is noticed, in all cases, that the requested rate is achieved only after a significant amount of time (between 30 to 40 seconds). This behavior is introduced by the AIMD shaper that linearly increases the maximal transfer rate when no congestion is noticed in the network. This behavior will cause several problems in traditional TCP traffic, due to the TCP slow start algorithm. The second thing worth noticing is the effect of intermediate shapers, as can be seen by the variation (decrease) in the raise of the curve when the throughput is increasing.

In Figure 3.9 the differentiation between classes for the same bit rate (128 Kbps in this case) can be observed. In order to observe the desired effect, the traffic is flowing from Node4 to Node1 (3 hops). The flows were started all at the same time and to each flow a traffic class was assigned. What can be observed is that real-time class starts at its maximum rate and lower classes take longer to reach the required throughput; it is also easily visible that the time to reach the required throughput increases with the decrease of the priority of the class. These results are consistent with the extended SWAN model, in where the real-time class does not have any shaper and initiates it service at the maximum bit rate, since it has absolute

priority. Best effort classes use the remaining bandwidth, which in this case its almost none.

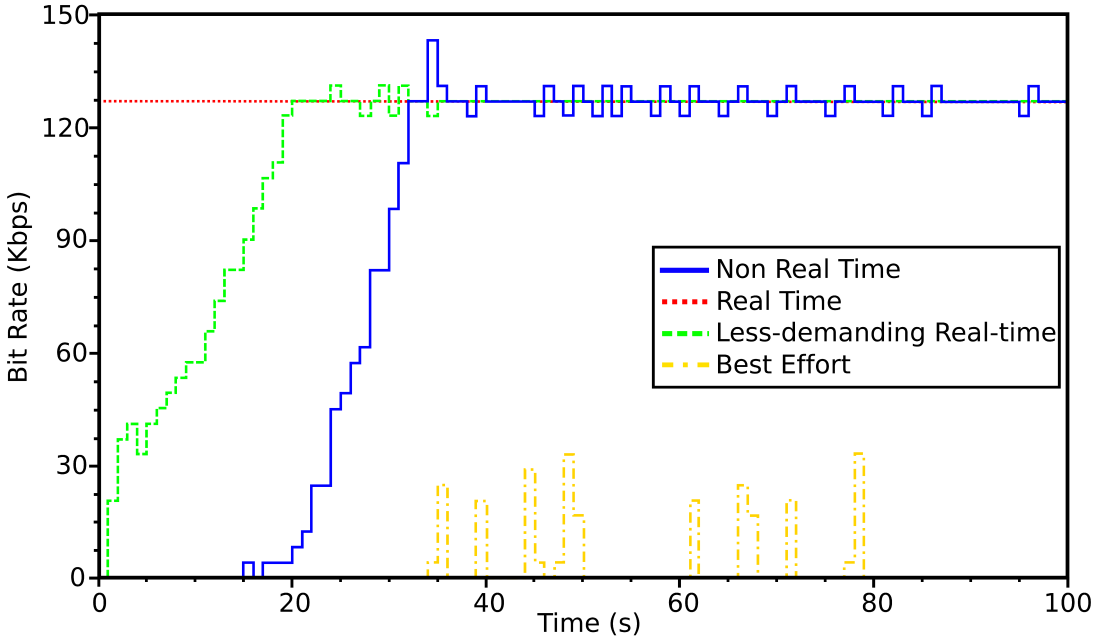

Figure 3.9: QoS initial setup differentiation for the first hop connection

In terms of delay and jitter values, for the real-time class, they are very similar to the values shown for Unicast routing in the previous section. For the other classes the delay and jitter values suffer from the strong influence of the shaping that is performed in all the nodes until the traffic reaches its destination.

The cumulative overhead for real time classes is of 2.11%, 1.20% and 0.67%, respectively, for the 64Kbps, 128Kbps and 256Kbps traffic profiles. Again these values are similar to the ones of multicast and unicast routing.

The presented results show that with the extended SWAN model it is possible to support service differentiation and regulation of flows. However, the number of hops in the ad hoc network has a large influence in both the maximum achievable throughput and in the time to achieve the requested rate. The system behavior is very dependent of the environment, frequently producing results that were not easily understandable in terms of service differentiation. The results also show that the real-time traffic class is the only one that does not compromise the network performance, for the target multimedia service types to be deployed, so the remaining integration test will only concentrate on the analysis of this traffic class, even when QoS modules are active.

3.5.5 Charging and Rewarding

Finally, the performance of charging and rewarding mechanism (PACP [2]), is evaluated. In this section we will only evaluate the overhead introduced by the charging and rewarding mechanism. In section 3.6 the measurements achieved with all the modules will be presented.

In the used scenario and performed tests (where the traffic flows from Node5 to Node1) PACP reports and PACP proofs generate almost the same rates of control bytes. However, PACP reports are sent in a burst every 37 data packets (each report contains the proof of 37 data packets), while PACP proofs are of constant size in all packets (48 or 88 bytes).

Overhead (%)	64 Kbps	128 Kbps	256 Kbps	Average
No ECDSA	10.98	21.96	43.90	17.15
With ECDSA	16.33	32.67	65.33	25.52

Table 3.8: Charging overhead (in Kbits) versus bit rate and usage of cryptographic mechanism

The results of the overhead measurements are present in Table 3.8. These results are dependent on the packet rate, which is due to the constant proof size and the constant number of reports issued per data packet forwarded. Another effect of the number of data packets in the network is the number of control bits introduced by PACP, which increases in a linear relation. The overhead is presented for two different situations: with and without security processing. Naturally the inclusion of security mechanism increases the overhead of the charging and rewarding protocol.

The elliptic curve digital signature algorithm (ECDSA) is the cryptographic algorithm used for the provision of the security. It is worth noticing that this choice will only be realistic if special hardware exists in the (low-power) ad hoc nodes, due to the high computational requirements of this algorithm.

3.6 Impact on the Usage of Ad hoc as Stubs Networks

In the previous section (3.5) the performance of the several components were evaluated and their results presented. In this section, the results are analyzed in the perspective of the final network arrangement, with all modules present and running. These results will be here analyzed from the perspective of their impact on the overall network performance, taking into account that all services are running simultaneously.

The overhead results presented in the last section for the auto-configuration functionality (1.44% in 64 Kbps of data) seems reasonable, since it is a essential feature on a stub ad hoc network. Further, since the overhead is a constant value, when the amount of data increases, the figure for the overhead of auto-configuration will be smaller.

Multicast routing is a feature of great value for a stub network. It permits resource optimization when delivering typical broadcast services (streaming multimedia contents). However, the results obtained in the performed tests have shown that ad hoc multicasting should be carefully considered when deployed in real scenarios. In a five hop network, the throughput available (Table 3.1) does not permit multimedia contents with high quality, but is good for low quality video. Albeit the delay values (Table 3.2) are of no great concern for the target multimedia services (a user can wait a little more for a streaming to arrive). High jitter values will have impact on the size of the cache that must be reserved at the terminals. The jitter values measured (Table 3.3) bring no worries. The most worrisome metric is however the packet loss, presented in table 3.4, since even when there is plenty of bandwidth to use, multicast traffic still show packet losses (even when unicast does not shown losses). Further search for the root of this problem reveals that it may be connected to the particular implementation of the MMARP protocol and is not linked directly to multicast in ad hoc networks in general.

The addition of unicast routing clearly introduces penalties in terms of delay, jitter (Tables 3.5 and 3.6) and overhead. Considering that voice communications require a jitter and delay lower than 50 ms and 150 ms, respectively, it is expected that all traffic profiles could be used, with the exception of 256 Kbps in a five-hop connection. Since unicast routing is a key feature in the ad hoc network, these results clearly introduce a figure for the maximum number of hops allowed in a stub ad hoc network. The 2.38% (for the same traffic profile) of cumulative overhead introduced by the auto-configuration functionality plus the unicast routing functionality is still acceptable, as shown in Figure 3.12.

The QoS services were designed to be compatible with multiservice networks, but the tests performed to evaluate them revealed that they are inefficient unless traffic belongs to the Real Time class (with no shaping). These results contradict the apparent adequateness of the proposed QoS solution for a multiservice network. The main reason for the bad performance outside the real time traffic class is due to the AIMD (Additive Increase Multiplicative Decrease) shaper. The shaper takes tens of seconds to increase the non real time classes throughput to the maximum, raising the delay and/or packet loss to unbearable levels. This problem is aggravated for the TCP protocol (due to the slow start mechanism), and consequently TCP

based services such as HTTP, which would not easily live under these conditions. These results show that only priority traffic (e.g., voice) is able to be usefully differentiated from other best effort traffic, with a cost of a little percentage bandwidth.

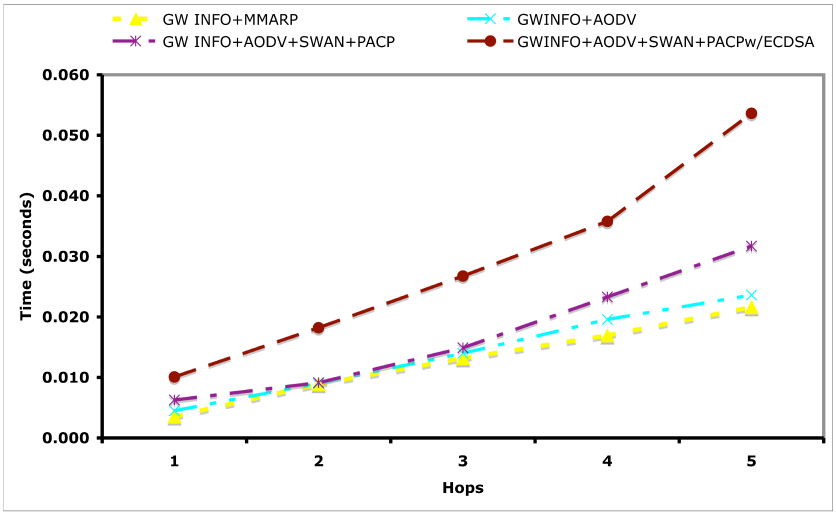

Figure 3.10: Cumulative delay for the 64 kbits traffic profile

In the complete multiservice network, with auto-configuration, unicast routing, QoS (only real-time traffic) and charging control active it is expected that the delay and jitter values would increase. In Table 3.9 and Table 3.10 the final results for the delay and jitter values are presented. As can be seen, without security methods, the values are slightly increased when compared with the ones using unicast routing, as a consequence of the packet processing and inclusion of proofs (as discussed early, real-time QoS impact is negligible). In addition, because PACP is implemented in user space, an additional context switch must be performed, as packets flow between kernel and user space. PACP directly controls buffering and queuing mechanisms, hence, when no cryptographic authentication is used, PACP control lead to more regulated traffic output, which slightly improves the performance of the network under congestion. The 256 Kbps test for 5 hops (Table 3.9) leads to heavily congested network (much larger than the available bandwidth), situation where PACP queue management actually leads to an improvement of performance.

Using the security functionality of PACP causes a negative impact on the one-way delay measured, significantly increasing it. For each packet that is sent into the network, it will be signed once by the sender, and then the signature will be verified in the receiver and in all nodes that have forwarded it. The ECDSA (Elliptic Curve Digital Signature Algorithm) signature is not expensive to produce (below 1ms), however the verification is. The test performed in the testbed show that the signatures verification takes between 3 and 5 ms on

Delay (ms)	64 Kbps	12 Kbps	256 Kbps
1 Hop	6.27	5.20	5.36
2 Hops	9.15	9.18	10.50
3 Hops	14.92	15.07	18.66
4 Hops	23.28	21.26	246.82
5 Hops	31.71	32.87	1106.62
Delay w/ECDSA (ms)	64 Kbps	128 Kbps	256 Kbps
1 Hop	10.08	10.09	11.16
2 Hops	18.20	18.24	18.39
3 Hops	26.74	27.00	33.37
4 Hops	35.77	35.48	592.29
5 Hops	53.62	67.15	1702.050

Table 3.9: Delay both with and without data authentication. Unicast routing, auto-configuration, QoS real-time and charging are active

the used hosts. Notice that these values are valid using ECDSA 163 bits, other key sizes will change this processing time. In the used 5 hop scenario, a packet originated in the last node with destination to the gateway will be verified 5 times and signed once, this gives a figure for the minimum delay that a packet will suffer when using ECDSA between 20 and 25 ms. The real value measured in the scenario is of 21.92 ms, which is according to the expected value (computed by reducing the values obtained without ECDSA to the values obtained with ECDSA for the five hop corresponding to the 64 Kbps traffic profile; see table 3.9)

Delay (ms)	64 Kbps	12 Kbps	256 Kbps
1 Hop	0.47	0.60	0.74
2 Hops	0.50	0.57	0.83
3 Hops	0.50	0.66	0.92
4 Hops	1.16	0.97	15.68
5 Hops	1.26	2.13	21.22
Delay w/ECDSA (ms)	64 Kbps	128 Kbps	256 Kbps
1 Hop	0.50	0.62	0.95
2 Hops	1.00	1.01	1.09
3 Hops	1.02	0.95	5.79
4 Hops	1.43	1.33	6.90
5 Hops	1.34	7.36	26.62

Table 3.10: Jitter both with and without data authentication. Unicast routing, auto-configuration, QoS real-time and charging are active

Figures 3.10, 3.11 and 3.12 show the results for delay, jitter and overhead, structured according to the incremental addition of modules, for some of the traffic profiles. The remaining traffic profiles show similar results and all considerations bellow are generally valid for all the tests performed. The graphics show that the main delay source is the charging

and rewarding mechanism, more specifically the secure authentication introduced. All other services do not significantly impact the delay of the packets in the network. The processing weight introduced by MMARP for multicast routing and by the signature verification of PACP significantly increases the variation of the delay achieved by the data packets, and hence, considering jitter, it is concluded that both MMARP and PACP with ECDSA introduce the higher penalties. The charging and rewarding services, including its security mechanisms, are also mainly responsible for the increase in the overhead.

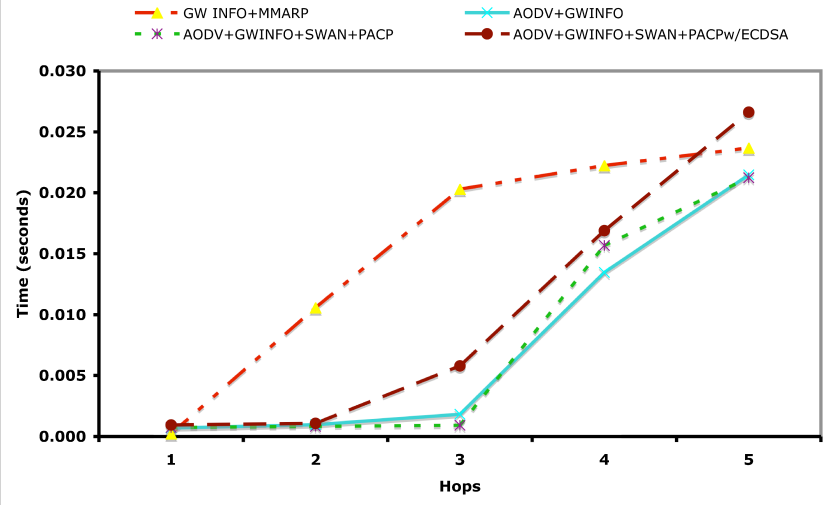

Figure 3.11: Cumulative jitter for the 256 kbits traffic profile

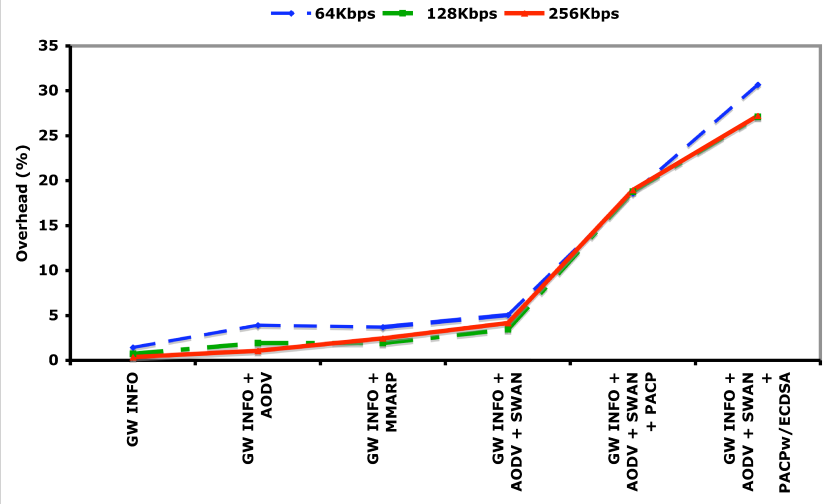

Figure 3.12: Cumulative overhead with the increase of functionalities

The inclusion of an ad hoc network in the operator environment requires that some significant control information is introduced in the network to enable the ”revenue” from the

ad hoc network deployment. However, the extra overhead introduced may be too much for the gained services (auto-configuration, routing, QoS and secure charging and rewarding) and therefore, the operator needs to balance these issues. The results may suggest that a different charging and control mechanism should be researched for commercial networks.

3.7 Conclusion

In this chapter it was shown, by means of the experimental evaluation of a real test-bed, the measured effects of introducing several functionalities of ad hoc networks serving as stubs for multiservice networks. The results obtained, overlaying multiple ad hoc network functionalities in a very simple scenario raise several concerns. For a start, the overall complexity of the software to be deployed in the nodes, and the number of potential interactions, makes the system quite prone to errors, and raises some interoperability concerns in a commercial environment with multiple software providers. Another concern is the large behaviour variability, mostly when routes are changing and/or QoS mechanisms are trying to regulate the network. In fact, during the test realization it was hard to obtain a stable, smooth behaviour of the testbed. Complete restarts between the several runs of a test were a very frequent practice. In addition, in small mobility scenarios, the effective usage of ad hoc networks seems not to go further than a couple of hops, as already seen in studies focusing single features.

The incremental addition of functionalities showed the tradeoffs that an operator needs to face when adding this kind of extra functions to its network, namely the impact that trust and QoS have on network performance. Imposing trust in communications rapidly increase the overhead in the network, and communications are throttled as soon as QoS regulation is taking place. These results show that a carefully scenario analysis should be developed before deploying ad hoc stubs in any multiple-service network: not all features will be effective in complex environments.

Using ad hoc as stub networks, the so-called ”extended hotspot scenario”, introduces an interesting concept and results show that the operators’ network coverage can be extended for a few number of hops. This number may vary according to the mechanism that the operator chooses to deploy, but will nevertheless be small if voice alike services are considered. A full functional stub network can support all features presented before, and still be able to maintain an acceptable performance with delays lower than the 50ms and jitters lower that 10ms for a maximum of two hops.

Chapter 4

Mobility Architecture for Ad hoc Networks

In this chapter it will be introduced and discussed the problem of performing mobility between an ad hoc network and other kind of network (e.g. Infrastructure, UMTS, etc.). A proposal for a solution will be described, and later analyzed. This proposed solution will be developed as a stand alone testbed and also integrated in the larger mobility testbed of the IST DAIDALOS II [60] European project.

The Daidalos II project is a research project that is working to define and validate the network architecture of future mobile operators. A key requirement for these networks is the support of ubiquitous access. Due to the current evolution of technologies it is the vision of the Daidalos II project that, to provide this ubiquitous access, users will have to access to the the services through a heterogeneous landscape of technologies, and through different types of networks, including mobile ad hoc networks (MANET) and moving networks (NEMO).

Daidalos II is defining a network architecture to provide ubiquitous access integrating heterogeneous access networks and providing seamless movement among them. The architecture will also support: mobility management splitting between local and global domains; identity based mobility management solution, through the independent and general managements of identities; integration both MANET and NEMO in the mobility architecture; host multihoming – the host owns multiple physical network interfaces and concurrently gets access through them; integration of ubiquitous and pervasive concepts for customized services to the users.

This chapter will address the support of MANETs integration developed under the

framework of Daidalos II. The architecture to be proposed aims at seamlessly support nodes moving from between infrastructure and ad hoc networks, maintaining its access to the Internet with the same quality. For this purpose the MANET needs to support routing integration, QoS support, security of routing and mobility with multihoming support. This chapter will focus only in the mobility part of the overall architecture.

The remainder of the chapter is organized as follows. Section 4.1 will introduce the problem of mobility in IP networks, along with the traditional solutions to solve the mobility problem. Section 4.2 describes the new approaches to solve the mobility problem. In section 4.3 the Daidalos project general mobility architecture is introduced and described. Section 4.5 will introduce the proposed solution for the MANET mobility architecture. Finally, section 4.6 will discuss the issues that the general architecture and its requirements could bring to the MANET operation.

4.1 Mobility in IP networks

When the Internet was in its early days (70s and 80s), very few computers and users were connected and using it. Additionally, in those times, computers were large, heavy and fixed machines, with a well defined set of users, so security was not a big problem. Mobility was also a problem not present at a time, so the developed protocols and solutions do not where engineered to cope with it. A consequence of those early times is that TCP/IP protocol stack, was made to work best with those conditions, resulting in a IP address with dual functionality: localization and identification of the owner computer. It provides the localization because an IP address represents the topological localization of the node inside the network, that is, the path to be followed by packets from another computer to it. The identification function refers to the fact that back then, an IP address was also associated with the same computer, identifying it fully.

Time passes, and the internet evolves from an academic research network to a global network of computers, with commercial interests. The number of computers connected to the Internet have had an exponential grow, which implies also a grow of the number of mistrusted users. Mobile communications began to rise (due also to the appearance of several wireless communication technologies), with a huge potential number of users and devices always connected to the Internet. These developments in the internet landscape began to raise problems and concerns that do not existed in the days the internet was created. Assuming the IP address of a computer represents its topological network location, and this specific node

is in a mobility scenario, like changing its point of attachment or performing roaming, the IP address must change. However, the address is also the identifier of the computer, so it should not change. The solution to these problems represents a compromise between these two contradictory concepts.

All the solutions that have been proposed to solve the problems mentioned above rely on *indirection* to work. Indirection is a technique that adds an intermediate layer to perform translation of one thing into the other. It is a technique widely used in the computer landscape, and also in networking, being DNS (Domain Name System) its widest spread usage.

The normal usage of indirection for mobility consists in the separation of the semantics attributed to the IP address into two planes: one plane for identification and the other for localization. The indirection mechanism then translates the identification of a node into its localization. Mobility in IP networks is a topic with a broad research, hence several solutions have emerged ([21], [41], [47], [6]). In the next sections two of the mentioned protocols will be more detailed, since they both will be part of the Daidalos architecture. Then, new mobility paradigms will be addressed, based on local mobility approaches. The IEEE 802.21 for media independent handovers is also briefly described.

4.1.1 Mobile IPv6

Mobile IPv6 [41] (MIPv6 for short) is based on Mobile IP [49] but takes advantages of the features of IPv6 addresses. MIPv6 allows that any IPv6 capable mobile node can change its point of attachment to the internet without requiring changes to the internet routing infrastructure, nor to other network entities, nor to the applications running in the mobile node. Mobile IP introduces several new concepts. For a start, it introduces the concepts of Home Link and Foreign Link. The Home Link is the link in where the MN's home subnet prefix is defined, e.g., the link in where the MN acquires its address; The Foreign Link or visited link, is the link that the mobile node visits upon movement, it is any link other than the home link. A consequence of using two different links is that MIPv6 utilizes two (at least) addresses per MN: one, known as the Home Address (HoA) that is the address of the MN in the home link and serves to identify the MN; the other address, known as Care of Address (CoA), that is the topological correct address and is used to deliver packets to the MN.

The HoA is an unicast routable address and is permanent to the mobile node, so that the standard IP routing mechanism can deliver packets destined to a MN HoA to its home link. The CoA is also an unicast routable address, and is attributed to a MN when this is

visiting a foreign link. Also the node can have several CoAs, but only one is used at a given time to send packets.

In order to properly function, MIPv6 introduces new network functional entities:

- Mobile Node (MN): Any node that can change its point of attachment to the network, without changing (and been reachable by) its home address.
- Home Agent (HA): A router belonging to the MN's home link with which the MN has registered its current care-of address. This router has also the responsibility of re-route all packets destined to the MN's HoA, to the MN's registered CoA.
- Correspondent Node (CN): any host in the internet communicating with the MN. The CN can also be a mobile node
- Access Router (AR): offers network connectivity and forwarding services to the MN

In figure 4.1 is represented an example network architecture for mobile IPv6, showing a MN, some CN, and the HA.

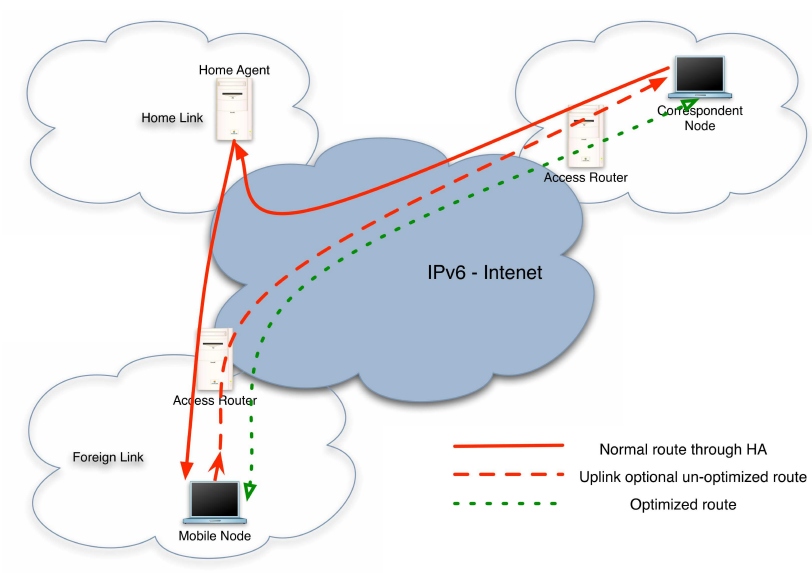

Figure 4.1: Mobile IPv6 example architecture

The basic functioning is as follows: When the mobile node reaches a foreign link, away from its home, it acquires (using IPv6 standard mechanisms) one or more core-of addresses. These care-of addresses have the network prefix (subnet prefix) of the foreign link, so as long as the MN stays in this location, the packets addressed to this care-of address will be

routed to the mobile node. After the mobile node gets its care-of address, it proceeds with the registration process, in which the MN register the CoA in its home agent, that is located in its home link. This operation will create in the HA a ”binding”, that is the association between a HoA and a CoA. The mobile node performs this binding registration by means of a ”Binding Update” message to the home agent. Upon reception of this message, the home agent responds with a ”Binding Acknowledgment” message. After this simple message exchange, a tunnel is created between the home agent and the mobile node. After this process is completed, the MN is immediately reachable by its HoA, through the home agent.

The communication between the mobile node and the correspondent node can happen in two different ways: using bidirectional tunneling or using ”route optimization”. Bidirectional tunneling does not require MIPv6 support in the CN, since the packets to the CN are naturally routed to the HA (home link) and then tunneled to the mobile node. Packets to the CN are tunneled from the MN to the HA (”reverse tunneling”) and then routed normally from the home network to the CN. This tunneling is performed using IPv6 encapsulation. To perform Route Optimization, the CN needs to have support for MIPv6, and requires that the MN registers its current binding at the CN, giving its current location. Packets from the CN can be routed directly to the CoA of the MN. The ability to route packets directly to the MN CoA allows that the shortest communication path between the MN and the CN to be used, allowing also a reduction in the overhead of the MN’s home agent and home link.

Mobile IPv6 solves the mobility problem by using, and identifying the mobile node with, two different address. One is used as the mobile node identifier and the other is the mobile node locator. The indirection is then performed at the home agent, which translates the identifier of the mobile node in its locator, or by the correspondent node itself when using route optimization.

4.1.2 Host Identity Protocol

The Host Identity Protocol (HIP) [47] provides a method for the separation of the locator and identifier role of the IP address, by means of the usage of a new cryptographic namespace, based on public key cryptography. The new namespace is the Host Identity Namespace. A name in this new namespace is the Host Identifier (HI), that is a statistically globally unique name, used to name any system with an IP stack. A system can have multiple identities (multiple HI), and some of them can be private and others public. In addition, a system may self-assert its own identity or may use a third-party authenticator.

The usage of the Host Identifier provides a very simple an straightforward way of providing mobility and multihoming, since upper (transport) layers bind to the host identifier, instead of binding to the IP address. The locators can then be changed seamlessly, without disrupting the ongoing communications. It is also possible for a single identity to have and manage multiple locators.

HIP adds two main features to the internet protocols: the decoupling of the internetworking and transport layers; and host authentication. The separation of the networking and transport layers are provided thanks to the usage of the Host Identifier by the transport protocols, as already explained. The host authentication is provided by the Host Identity, that is composed by a pair of public and private keys. The public key is the Host Identifier (HI); the host is only referred by this public component. The possession of the private key defines the Identity itself, so if the private key is possessed by more than one node, the Identity is considered a distributed one. Host Identities are never used directly in internet protocols. Instead, their public parts (Host Identifier) are distributed and stored in DNS and LDAP directories, so that they can be used in the HIP Base Exchange.

The locator function in HIP is performed by the Host Identity Tag (HIT), or Local Scope Identifier (LSI), so that the internet protocols can refer to the Host Identifier. A HIT is a 128-bit representation of the HI, created using a cryptographic hash. The usage of the HIT over the HI has two main advantages: its fixed length makes it easier for protocols coding; and it presents the identity in a consistent format to the protocol, independent of the cryptographic algorithm used. In the HIP packets, the HIT represents the sender and receiver of the packets, so a HIT should be unique in the hole IP universe, during its life time. The LSI is a 32 bit equivalent of the HIT, being also a representation of the HI. It has the purpose of facilitating the usage of HIP in existing APIs and protocols, namely providing compatibility with IPv4. The LSI advantage is its reduced size over the HIT, but its local scope, presents its main disadvantage.

The indirection mechanism introduced, by means of the Host Identity, clearly performs a separation between the location and identification function. The IP addresses, represented by the HIT/LSI, continue to function as locators, at the same time, the Host Identifiers serve the purpose of identify a node or group of nodes. The support for end-host mobility and multihoming comes naturally with the indirection mechanism introduced. The mechanism provided by HIP permits that several IP can be linked together, making multiple IP addresses correspond to the same Host Identity. When one address becomes unusable, or a more preferred address becomes available, the existing transport associations can be easily moved to

another address. Address changes during on-going communications are straightforward. The mobile node must send a HIP re-address packet to inform the peer of the new address(es), and the peer must verify that the MN is reachable through these addresses.

To ensure that nodes are reachable during, and after mobility, dedicated mechanisms are needed. HIP has a rendezvous mechanism that enables a mobile node to start a HIP exchange with another peer [45]. The MN keeps the rendezvous server continuously updated about its current IP address(es). Upon a communication start, the initiator must send the first HIP packet to the rendezvous server (whose address must be obtained from the DNS). The rendezvous server then forwards the first packet to the correct address, allowing the HIP base exchange to flow normally afterwards.

4.2 New mobility paradigms

In the previous section, the need to perform mobility was introduced and explained, as well as the normal approaches to solve it. Both approaches presented follow the philosophy of host controlled movement, with assistance from the network. These approaches are good to perform mobility, but require the intervention of the mobile node every time movement has taken place, which will have a negative impact on the network performance, mostly on large networks. For example, mobile IP has well-known and widely studied deficiencies in performance. These issues have been tackled by recent research teams, producing solutions that favor the optimization of movement by exploiting their local nature. The IETF NetLMM Working Group [72] is one of the research groups currently studying the localized mobility management paradigm, converging to achieve a standard protocol.

Another issue currently under research is the handover control. Both of the detailed techniques in the previous section follow the philosophy of host controlled movement, completely managed by the mobile node. New solutions for detecting the need to handover, control the handover realization, and all other actions related to handover are being researched. These approaches tend to completely revoke the mobile node control over the mobility functions, letting the network take care of them, or split the mobility control and decision responsibility between the network and the mobile node. The IEEE 802.21 Media Independent Handover [35] in-development standard is one of the solutions proposing that the network can control the handover functions of the mobile node, among other things. Next sections address local mobility management protocols and IEEE 802.21 for media independent handovers.

4.2.1 Local Mobility Management

All Local Mobility Management protocols rely on the concept of local and global domains. The exact definition of local and global vary from protocol to protocol or architecture to architecture, mostly on the size of the local. The general purpose, however, is the same for all protocols and architectures.

A global domain generally is considered to be the Internet. The concept of global domain is used to express the situation when the global mobility protocol could no longer work, and then the global mobility protocol (as MIPv6 or HIP) is required to perform the mobility signaling. A local domain can be either an administrative region or a physical region on the network. For example, a local domain can be the network zone formed by all the access routers that use the same core router to reach the internet (see figure 4.2), making all movements outside the scope of that router global. Another example for a local domain can be an entire operator network (comprising several access routers and core routers).

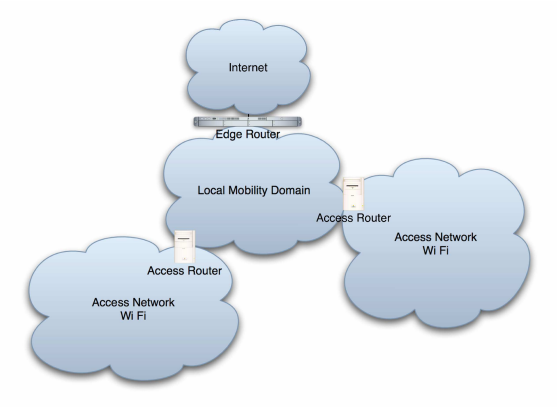

Figure 4.2: Local mobility domain examples

Localized Mobility Management protocols are divided in Host Based protocols, where the host takes care of the signaling, and Network Based protocols, where the terminal does not require to perform any signaling since is the network that controls mobility.

4.2.1.1 Host Based Localized Mobility Management

Local Mobility Management is not new. The first proposals were host-based, leaving to the host the task of controlling all the mobility signaling, making them aware of both local and global signaling protocols. Between the most relevant previous protocols were Hierarchical Mobile IP and Cellular IP.

Hierarchical Mobile IP [53] for IPv6 (HMIPv6) introduces a new Mobile IPv6 node, called the Mobility Anchor Point (MAP), that is used and located at any level in a hierarchical network of routes, including the Access Router. Traditional Mobile IPv6 requires that the mobile node sends a Binding Update to all Correspondent Nodes to which is communicating, and to the Home Agent, every time any movement is performed. Combined with the authentication mechanism, MIPv6 tens to be very heavy on the network and not very fast, when there exist several peers and active communications. This issue is more serious when using wireless links. The usage of the MAP will limit the number of Mobile IPv6 signaling that travel outside the local domain, providing also a solution to the referred problems. The mobile node will only send Binding Updates to the MAP (and not to the HA and CNs), and only one Binding Update message needs to be transmitted before the MN can resume traffic reception and transmission.

The MAP acts like a local Home Agent, enhancing in way the performance of MIPv6 while minimizing the impact on MIPv6 and other IPv6 protocols. The local domain is composed by all the Access Routers that the MAP controls, making all handovers between that set of MAPs optimized. Because HMIPv6 is a host based solution, it requires that the mobile controls both the local and global mobility signaling. In addition, HMIPv6 introduces another tunnel over the air, in case of wireless links, causing the data traffic to be inside two tunnels. The first tunnel is the original MIPv6 tunnel between the MN and the CN or HA; The second tunnel is established between the MN and the MAP.

Cellular IP (CIP) is a local mobility management protocol more close in its concepts and operation to real cellular networks, like GSM, than to traditional IP networks. The mobility in the local domain is handled by CIP, and in the global domain by MIP. In the Cellular IP architecture, a node called gateway is the node that provides connectivity of the local domain to the internet, making it the edge router of the domain. This gateway periodically broadcast its address to the local domain. When a MN enters the local domain, it listens these broadcast messages and then registers itself on its HA and CN with the address of the gateway. Upon registration of the MN, the CIP network, that can be composed of several routers (special routers) and switches between the MN and the gateway, learns the address of the MN (the HoA), creating a host route from the gateway towards the MN. This route is updated based on the uplink traffic sent by the MN, or by special signaling messages. Each router learns from the uplink traffic which is the next hop to which send the traffic destined to the MN. The uplink traffic is forwarded hop-by-hop, towards the gateway, regardless of the destination.

Much like cellular networks, Cellular IP has power consumption saving mechanism.

When a node does not send or receive traffic during a predetermined period of time, the MN enters an IDLE state. Entering this IDLE state could cause the exact location of the node to be inaccurate. It is up to the gateway to periodically broadcast paging messages in order to regain knowledge of the IDLE nodes location, causing also the MN to enter an ACTIVE state.

Cellular IP is a host based solution, since it requires the MN to explicitly register in the gateway when it arrives to the local mobility domain. This procedure allows the node to discover which CoA to register in its Home Agent and to the network to gain knowledge of a new MN to which it has to construct routes.

4.2.1.2 Network Based Localized Mobility management

As referred in the introduction to this section, Network Based Localized Mobility Management protocols, are a topic of current research, including also standardization effort in IETF. The basic philosophy of network mobility management is the relocation of relevant mobility management functions from the mobile node to the network.

The NetLMM (Network Localized Mobility Management) basic requirements and problem statement are already defined in the NetLMM WG of IETF by [43] and [44]. The adopted approach requires that the network learns through standard terminal operation (such as router- and neighbor discovery or by link-layer support) about the terminal movements inside the local domain, so that it can coordinate routing state updates without any mobility specific support from the terminal. Following this approach enables the support for hierarchical mobility management, since the mobile node signals is location update to a global mobility anchor point (such as a Home Agent) when it changes LMD and, inside the LMD, mobility is supplied by the network to the terminal without any type of terminal originated signaling.

The basic architecture of NetLMM is presented in Figure 4.3, where it can be observed the entities involved in the protocol operation. The Local Mobility Anchor (LMA) is the router that establishes the border between the local (NetLMM) domain and the global (core) domain. The Mobility Access Gateway (MAG) is the Access Router for the MN. All intermediate nodes between the MAG and LMA are NetLMM unaware, reducing this way the signaling needed to maintain mobility information, and resource consumption and overhead.

The area between the LMAs and the MAGs is where NetLMM operates. For every mobile node, a tunnel is created between the corresponding MAG/LMA pair. All traffic destined to the MN is sent through the tunnel to the MAG, and then the MAG delivers it to

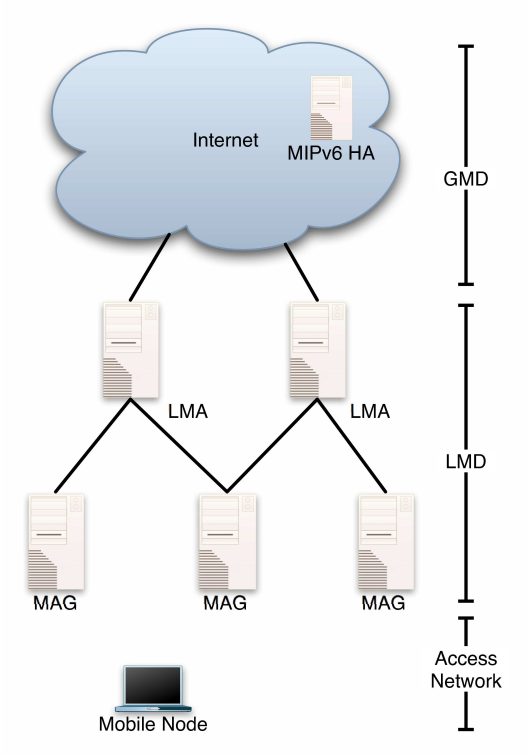

Figure 4.3: NetLMM basic architecture

the MN. The upstream traffic follows the opposite procedure, the MAG receives the traffic and sends it through the tunnel to the LMA, which sends it to its destination. These tunnels are normal IPv6 in IPv6 tunnels.

When the MN moves from one MAG to another, the new MAG detects its presence and signals the LMA. A new tunnel for the MN is established between the LMA and the new MAG, and the old tunnel is removed. During this procedure the MN does not know of the movement, maintaining its IPv6 address (CoA) whenever it is managed by the same LMA.

The basic operations described above are from the first NetLMM WG draft [27]. The solution presently under discussion and development in the IETF is Proxy Mobile IPv6 (PMIPv6) [33]. PMIPv6 is a network based mobility management protocol based on MIPv6, reusing some of its functionality, namely the Home Agent function. The operation of PMIPv6 is similar to the already defined, however some differences exist.

In PMIPv6, the Local Mobility Anchor (LMA) is the the home agent for the mobile node in the PMIPv6 domain. It has all the functionality of a MIPv6 home agent, and also the required extensions to support PMIPv6 operation. The MAG remains the same, but gains additional functionality, the Proxy Mobile Agent (PMA). Its responsibility is to track the MN

in its link, and perform the signaling with the LMA, on its behalf (proxying).

In a PMIPv6 domain, the network will ensure that the Mobile Node will always be in its home network, ensuring also that the mobile node can always obtain its home address on the access link, using any of the address configuration procedures. As the mobile node perceives it, the entire PMIPv6 domain is its home link. PMIPv6 also establishes a tunnel between the MAG and the LMA, so that traffic is properly routed to and from the mobile node. Mobility is also handled in a similar way to what was already described.

Proxy Mobile IPv6 has some additional features. Each mobile node has an identifier, called MN-Identifier. This identifier is communicated to the LMA upon the MN's authentication in the network. Using this identifier the LMA can obtain the MN specific policies recurring to a policy store, such as the Authentication, Authorization and Accounting (AAA) infrastructure. This policy information is used to emulate the MN's home network on the current access link. Every detail, since which address configuration method is used, to the IPv4 support, is personalized recurring to the policy information.

The interaction between the MN and the MAG is performed recurring to standard IPv6 operations. These operations are of great importance to the proper operation of PMIPv6. When a node enters a PMIPv6 domain, it will send a Router Solicitation message, containing its MN-Identifier. The MAG will then send a Proxy Binding Update request to the LMA, which will contact the policy infrastructure to obtain the MN's related information. Next, the LMA will send a Proxy Binding Update acknowledgment to the MAG indicating a proper authentication and containing the Proxy Care-of Address. Upon receiving this message, the MAG will respond to the MN with a Router Advertisement containing the MN Home Network prefix and, at the same time, establish an IPv6-in-IPv6 tunnel with the LMA, so that traffic for and from the MN can be properly routed. After this bootstrap operation, the node can move freely in the PMIPv6 domain, which is presented to the MN as the same link. It is up to the MAGs to detect the MN movement and update the state of the LMA and other MAGs so that traffic can always reach the MN.

4.2.2 IEEE 802.21 Media Independent Handover

The IEEE 802.21 Media Independent Handover standard [35] aims to establish a standard that provides link layer intelligence and other related network information to upper layers in order to optimize handovers between heterogeneous networks. This standard is still in development inside the IEEE. The IEEE 802.21 standard currently considers the both wired and

wireless IEEE 802 family of standards, Third Generation (3G) Partnership Project (3GPP) and 3G Partnership Project 2 (3GPP2),

The main purpose of the IEEE 802.21 standard it to enhance the user experience of mobile devices by supporting handovers between heterogeneous networks. Handover support is provided for both, mobile and stationary users. For the later, handovers may occur due to several causes, most common causes are sudden change in wireless/radio conditions of the mobile node; usage of applications that require a higher rate data channel; or also degradation in a the network performance, that makes it less attractive in comparison to other neighboring networks. In any case the service continuity has to be maximized during the handover.

Cooperative use of network information is also supported by the 802.21 standard. This information is available at the mobile node and within the network infrastructure. In the mobile node, since it is well positioned to detect the available surrounding networks. The network infrastructure is used to store overall network information, such as neighborhood cell list, location of mobile nodes and also higher layer service availability. The mobile node and also the network points on attachment (base stations, access points) may also be multimodal – support for multiple radio standards and support for simultaneous connections on more than one radio standard.

The 802.21 standard consists on the following elements:

- A framework that enables seamless service continuity while a mobile node switches between heterogeneous link-layer technologies. This framework presents the Media Independent Handover (MIH) reference models for different link layer technologies.
- A set of handover functions below the protocol stack of the network elements, provided by a new entity called the MIH Function (MIHF). The MIHF provides the following services:
  - The Media Independent Event service that detect events and delivers triggers to both local and remote interfaces.
  - The Media Independent Command service that provides a series of commands to for the MIH Users to control link states that are relevant to handover.
  - The Media Independent Information service that provides the information model for query and response, thus enabling more effective handover decision making across heterogeneous networks.

- A media independent handover Service Access Point (MIH_SAP) and associated primitives, to provide the MIH_Users with access to the services of the MIHF.
- The definition of new link-layer SAPs and associated primitives for each link-layer technology. These new primitives will help the MIHF to collect the link information and control link behavior during handovers.

Figure 4.4 shows the location of the MIHF within the mobility protocols stack on a MN or network entity and the medium layers. In this picture it can be observed how the IEEE 802.21 standard can be media independent. As it is clearly visible, the upper layer protocols only have to support one interface to interact will all the supported technologies.

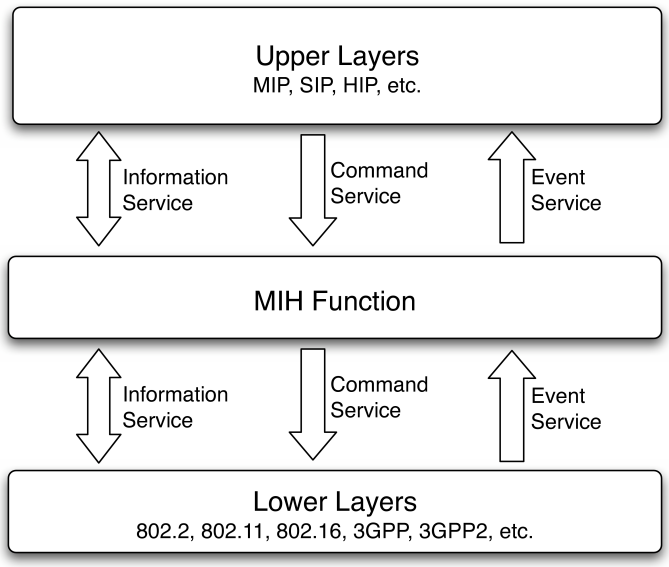

Figure 4.4: MIH services and their initiation

The MIHF does not only interface with the lower layers of the equipment it is located, but also with remote equipments in the network, recurring to L2 or L3 message exchanges. In figure 4.5 there is the representation of the available SAPs in the IEEE 802.21 standard.

The IEEE 802.21 standard is not a replacement for upper layers mobility protocols such as Mobile IPv6 or HIP (see section 4.1), but aims to help the work of these protocols. In addition, current 802 family protocols do not support handover between different types of networks, nor across different subnets. IEEE 802.21 will provide traditional mobility protocols with triggers and information useful to accelerate not only the handover execution, but also the need to perform handover, all in a media independent way.

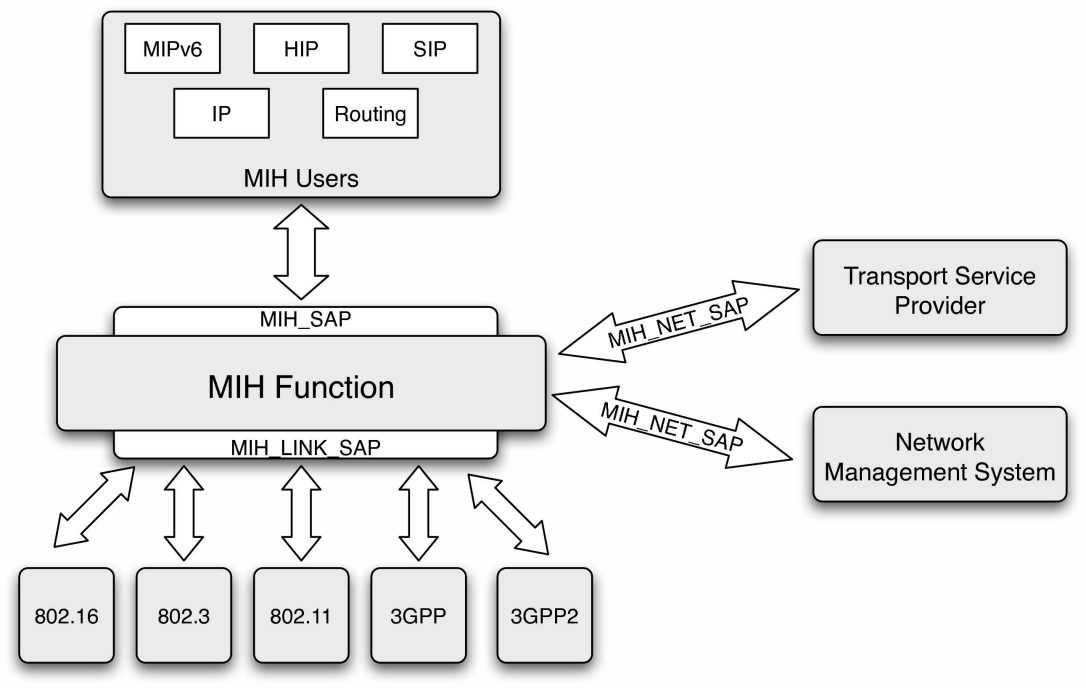

Figure 4.5: MIH Function communication availability

4.3 General Mobility Architecture

The Daidalos [60] mobility architecture, in which the MANET mobility architecture fits and bases on, is designed to cover the needs of the operators in the future. This architecture is based on the currently ongoing work of the Standard Development Organizations, such as the IETF, which split the mobility problem in local and global mobility. As explained in the previous section, in this concept of mobility separation, the network is divided in several local domains connected via a core network. Inside each local domain, protocols such as PMIPv6 are used to handle mobility. The global mobility protocols are only used when the node moves across local mobility domains.

Recurring to the mobility management split has several advantages. Local mobility domains (LMD) are independent of global mobility domains (GMD), as a result local mobility protocols (LMP) are also independent of global mobility protocols (GMP). A direct consequence of this independence is that operators are free to chose their preferred LMP or even skip its usage completely, leaving mobility management inside the LMD to the GMP. The reduction of signaling gained both outside the local domain, as between the terminal and the access network (over the air) is also a gained advantage. Another advantage is that a LMD could be constituted only by a certain access technology, hiding from the rest of the network

the specificities of that technology mobility management. This transparency helps the integration of technologies such as WiMAX (IEEE 802.16), 3GPP and also different types of networks such as NEMO and MANET.

Figure 4.6 shows the general network architecture, with one cloud for each of the separated technologies. As can be seen, the mobile terminals are capable of perform multihoming, and the network will provide support for it.

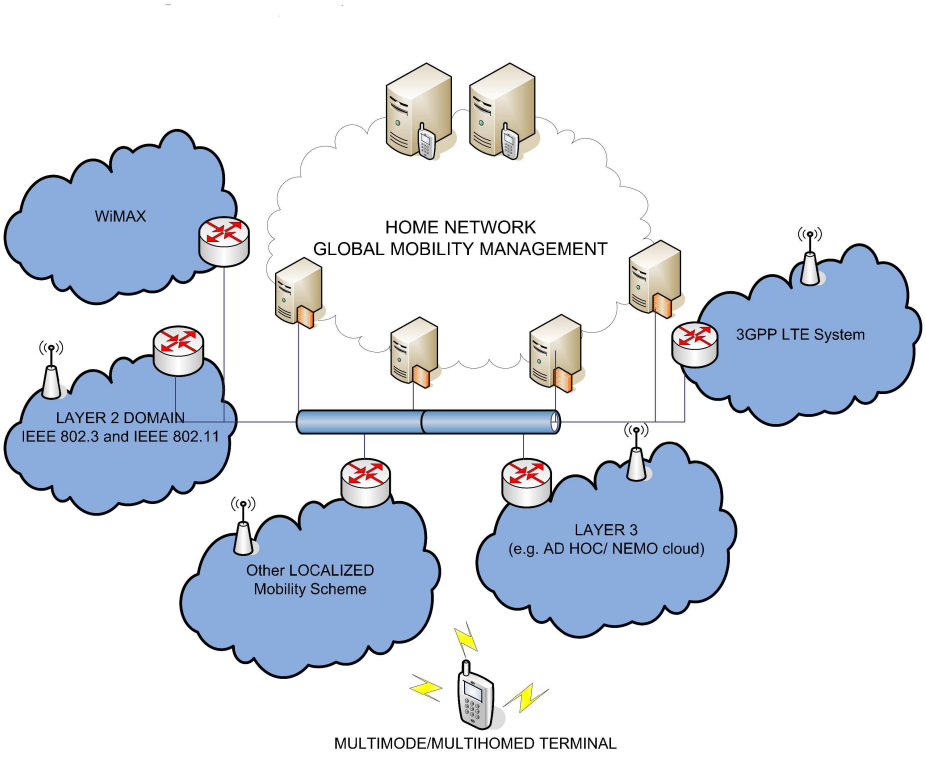

Figure 4.6: General network architecture

Mobility architecture was designed in order to provide the best service to the mobile terminals, providing also service continuity for whatever access technology the terminals use and for whatever technology change (handover) the terminal performs. However, there are various types of terminals. Terminals with only global mobility protocol support and no other enhanced mobility function will only be provided with transparent mobility inside the local domain, and can equally move between local domains using the GMP, but due to their lack of extended functionalities, their handovers will not be optimized, and therefore some service interruption may occur. When terminals have the enhanced mobility functions, such as support for IEEE 802.21 and local mobility protocol, support for seamless handover is more straightforward.

The enhanced mobility functions are a cross layer interface derived from the IEEE 802.21 which will be present in non-legacy LMD. This functionality is provided both at the terminal and at the network side, and provides the LMP with triggers and information such as movement detection for proactive handovers.

The general mobility architecture was designed taking into account the following requirements:

- **R1**: Access Network Operators can implement their own mobility solution.
- **R2**: Minimize complexity in the terminal.
- **R3**: Efficient use of wireless resources.
- **R4**: Reduce signaling overhead in the network.
- **R5**: The solution must be security friendly.
- **R6**: Seamless handover support.
- **R7**: Multihoming support.
- **R8**: Scalability for routing.
- **R9**: Minimize network side notes modifications.
- **R10**: Support for heterogeneous networking.
- **R11**: The solution must be QoS friendly.

4.4 MANET support for General Mobility

In the next sections the problems and solutions related to the deployment of a MANET network in the previous general architecture will be presented. The next section will outline the main issues and problems faced during the development of the architecture. In the remaining sections each issue will be explored in more detail.

4.4.1 Problem statement

The integration of the MANET on the Daidalos general architecture must comply with the requirements established for the general architecture. The resulting MANET architecture has

to be fully integrated in the general architecture, making MANET as transparent as possible to the remaining architecture elements. The key idea is that the difference between a terminal that supports infrastructure mode and a terminal that supports MANET will only be a little more software, allowing an MN to handoff between infrastructure and MANET without trouble.

The main issue faced is that, although a great deal of concern has ben taken to develop a mobility architecture that supports heterogeneous networks, the final result is still oriented towards infrastructure-based terminals. The MANETs is the only multihop technology been integrated in the architecture, so it has to adapt to the others architectures requirements.

Mobile Ad hoc Networks will be integrated in the same local mobility domain as regular 802.11 infrastructure networks, forming a heterogeneous LMD. In the LMD, the Local Mobility Protocol is network based, that means that no software or modules are required in the regular infrastructure Mobile Node to support mobility. In MANET, all the ad hoc network has to cooperate so that the MAG can know when a MN leaves the network and joins a new one (performs handover), and so the necessary modules have to be created.

Terminals that have the enhanced mobility functionalities, rely on IEEE 802.21 to trigger and signal the movement of nodes inside the network. Since 802.21 is designed to support only infrastructured networks, the MANET will have to adapt in order to support and comply with IEEE 802.21 standard (see section 4.4.3)

Before executing the (soft) handover, the MN has to acquire several information about the available networks, in order to choose the best destination network available. Establishing a parallel, in the infrastructure mobility solution no L3 (but QoS) information is considered by the decision logic to determine the best handover destination. In MANET L2 information is not sufficient to determine the best handover target, since the topology of the destination ad hoc network has a direct impact on the achievable performance. The information about topology (and related issues) will have to be obtained from the ad hoc L3 protocols. This information is then fed to the decision logic so that the best handover destination can be calculated. Having this kind of extra information will also improve the choice for the target technologies of the handover. More details can be found in section 4.4.4.

After handover execution and handover targets selection, multihoming is the next problem that needs to be considered, with a solution contemplated in the final architecture. In the Daidalos MANET, multihoming is different from the infrastructure solution: since a MANET can have multiple gateways, the MN can distribute flows by them. This distribution of flows

has to take into consideration topology characteristics of the network (i.e. the distance, in hops, of such gateways). In addition, multihoming can be performed with only one gateway, but recurring to multihop paths towards the only one gateway, which will add some extra requirements to the multihoming management: multihoming will have to consider the ad hoc metrics, such as number of hops to the gateway, to perform the best traffic distribution.

The issues discussed above were the those that have most influence in the design of the mobility architecture for the MANET. The remaining features (QoS, ad hoc routing) of the architecture were adapted to the mobility driven solution. However, this remaining aspects will not be discussed in detail, unless the mobility architecture has a negative impact on them.

4.4.2 Support for the Local Mobility Protocol

As stated in previous sections (section 4.2.1 and 4.3), the localized mobility management concept envisions that the mobile node can run unchanged and local mobility operations are performed by the network equipments. This requirement implies that it is the responsibility of the network Access Routers (ARs) to detect the terminals attachment and detachment to the access networks, in order to trigger the necessary mechanisms to perform the terminal's movement management.

In infrastructure networks, the movement detection (attachment/detachment) can easily be done recurring to the standard mechanisms present in the IPv6 protocol stack (such as the ICMPv6 neighbor solicitation and neighbor advertisement mechanism). However, in the multihop environment provided by the MANETs this standard mechanism cannot be used, and thus has to be replaced for either an explicit signaling performed by the mobile node, or by the help of explicit signaling done by the terminal's neighbors in the MANET.

Since the terminal has to perform explicit signaling upon handover to maintain QoS reservation and to get the needed authorization to perform handover, the most reasonable solution is that terminal also notifies the MANET Access Router (gateway, GW) every time it performs an attachment. In this case, the local mobility protocol will be responsible for the detachment at the old gateway.

In order to support the local mobility protocol operation in a transparent way, the proposed solution has the following characteristics:

- When the node joins a new network (or finds a new gateway at the current network), it has to register explicitly, in order to configure its address and also to signal the GW of

its arrival;

- The local mobility protocol in the core network will run unchanged, and thus, not knowing the difference between a MANET terminal and any other terminal;
- It is the local mobility protocol responsibility to notify the old access router of terminals detachments, so that the network resources can be correctly managed.

The crucial part of the support for the local mobility protocol is that the mobile terminal will have to directly signal the gateway during the address configuration phase, so that the LMP engine at the access router can be correctly triggered. Altering the MN to support local mobility is preferred over alter the local mobility management scheme in order to support MANET.

4.4.3 IEEE 802.21 support in MANET

The IEEE 802.21 standard has not been developed with multihop networks as objective; without adaptations, it will not work or be useful to ad hoc networks. The degree of integration intended for the MANET network in the overall architecture implies that the mobile node will not be used exclusive for MANET, instead the same terminal will be used to access regular infrastructure networks, 3GPP networks, etc. possibly at the same time. This requirement forbids strong alteration of the MIHF at the mobile node, since it is shared among all other technologies. The solution is to adapt and hide the MANET to the IEEE 802.21 standard, making it act like a regular single hop technology.

Another aspect that prevents the direct usage of the IEEE 802.21 standard in MANET is the information transport. IEEE 802.21 supports remote users that can subscribe to events that occur on the mobile node. In addition, the commands can travel through the network from and into the mobile node. In the IEEE 802.11 subfamily of technologies, these messages are transported as L2 messages (recurring to management frames) when in infrastructure mode. This L2 transport is useful since the MN does not need to have an IPv6 address to communicate to the Access Router, which facilitates the communication exchange between the mobile node and the access router, also, without the need to disrupt the upper layers. However, L2 messages transport does not work in the MANET, since all the routing and other basic services are only offered at L3 (i.e. IPv6) layers.

In order to make the less changes to the MIHF and to provide proper transport for IEEE 802.21 messages, the best solution is to hide the MANET nature from the MIH. This

is achieved by adding an extra layer between the MIH and the wireless driver. The MIH layer will see 802.11 mode ad hoc as a different technology of 802.11 mode infrastructure. MANET is a L3-only technology, and it will be presented that way to the MIH layer. All IEEE 802.21 messages that need to leave the mobile node in order to reach their destination in the network, will be transported as payload of IP messages, and therefore correctly routed by the normal ad hoc routing protocol in use.

For a full IEEE 802.21 standard integration in the target MANET there is still an issue to be solved: internal mobile node link events. In normal operation all events perceived by the 802.11 driver (access point found/lost, signal increase/decrease, etc.) are fed to the MIHF recurring to link_events. When in ad hoc mode, these events have no meaning to the MIHF (e.g. a new AP can really be a new ad hoc neighbor detected). In addition, the events are then fed to all MIH users that have subscribed to link status events. In a very large network this behavior will *spam* the MIH Users with messages that have no practical meaning. As discussed previously, the events that are relevant in the MANET are produced due to L3 protocol operation, such as the ad hoc auto-configuration module and routing protocols. These L3 events (e.g. a LinkDetected event when a new gateway is found five hops away) are not possible to be produced by the drivers, so they have to be fed to the MIHF by other means.

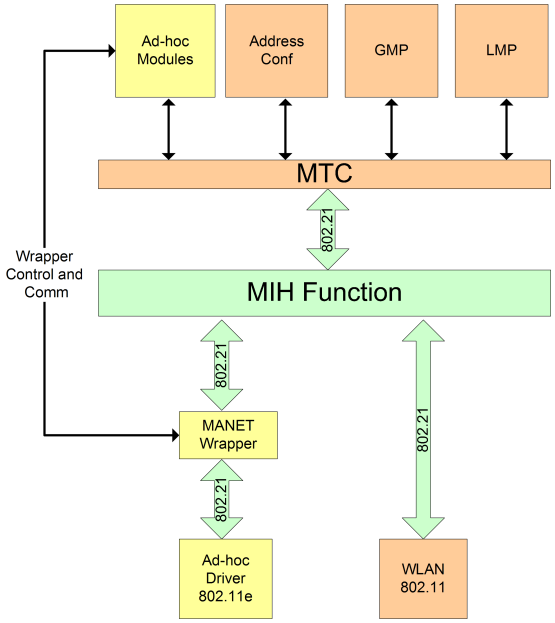

Figure 4.7: Simple MIH architecture in the Mobile Node

The proposed solution, basically sketched in figure 4.7, is to introduce a new entity

that operates between the 802.11 driver and the media independent function at the mobile node. This entity will act as a wrapper (its name is MANET Wrapper), hiding the MANET operations from the MIH, appearing to the MIH as a driver and to the driver as the MIH. MANET Wrapper will also help presenting the MANET as new technology to the MIHF and will also have an important function in the multihoming controlling logic. The MANET controlling modules will be responsible for controlling the MANET Wrapper, and through it send the important events to the MIHF.

Summarizing, the MANET Wrapper will be the central entity that will provide support for IEEE 802.21 standard to the MANET and provide MANET support for the IEEE 802.21 standard. MANET will benefit since it will be integrated with a new technology that provides mobility support in heterogeneous networks. This integration will make MANET one more option for usage in large networks, along with traditional infrastructure networks. The IEEE 802.21 standard not be aware of the MANET issues (multihop links), and will only receive from the MANET wrapper meaningful information, as if the node is connected directly to the gateway.

4.4.4 Handover candidates discovery

Before performing any handover, it is needed to know where to perform the handover. The handover target can be obtained in two ways: one it that the mobile node is informed by some entity in the network, recurring, for instance, to the IEEE 802.21 Information Service; the other method is an active scanning by the mobile node. For the first method to work, the network has to know the exact location of the mobile node in relation to the network's access points, which is difficult to achieve in a ad hoc network. Using active scanning the MN will only consider as handover targets the networks that it can reach, since they are the results of the scanning. Scanning for surrounding networks are a better option for handover target discovery.

The proposed candidate discovery process is divided in two phases: in one phase it is performed an ordinary wireless scan, and in a second phase the scan results that correspond to ad hoc networks are validated. The scanning performed in the first phase is the same scanning procedure done to determine the handover candidates for the infrastructure mode. This process returns a list of available networks; each entry on the list consists on a tuple (ssid, channel, mode, etc) that holds the necessary information to choose the best PoA for the handover destination in infrastructure mode. However, the information gathered is not

sufficient to choose a good ad hoc PoA since it is required to know what kind of routing protocol is the network running, to what operator belongs the network, if it is possible to reach at least one gateway, and what is the (IPv6) address of the gateway. All this information is gathered from the auto-configuration module.

During the validation process, for each ad hoc network, the terminal will have to associate to it and wait for the periodic auto-configuration message to arrive (it can also be requested, which is recommended to reduce idle times). After receiving this message, the terminal will extract the needed information so it can be handled by the decision logic. This approach has some advantages. First, it is independent of the driver used: it can perform the scanning and validation with whichever WIFI card the terminal has, and not only the ad hoc one. Second, it will not require using a special (and possibly non compatible with legacy nodes, and thus the 802.11 standard itself) ad hoc wifi driver.

The separation of the handover candidates discovery in these two phases also fits better in the 802.21 architecture envisioned in the previous subsection (Section 4.4.3). The MANET Wrapper (MW) (Figure 4.7) is extended to perform two tasks now: support for the 802.21 protocol in order to deliver the correct events to the MIH layer, and to coordinate the validation phase of the Candidates discovery. There are still some issues to consider in the candidate discovery scenario. This process will interfere with the normal communications the node is performing, if for the scanning purposes only one wireless card is used. To circumvent this problem, the design of the final solution was made to not having for requirement any specific number of wireless cards; this way we can assure it will work in any given scenario. If the MN only has one wireless card, all the phases of the network scanning will be performed in the physical card, and in consequence, some performance drawbacks will have to be expected. In the case the ad hoc modules can have to their disposal two (or more) wireless cards, then only one of them can be used to perform the scanning and management tasks, and the other for the actual communications the node is carrying. This second approach may be absurd, but in the not so distant future there will be available wireless cards that are multi-radio, and thus supplying all the needs established now. For the time being and for demonstration purposes, a driver can be built in a way that it controls two physical cards in order to deliver the functionality only present in the future hardware.

4.4.5 Virtual identities and Virtual interfaces

In Daidalos architecture there are multiple identifiers used at different levels. Like in traditional IP networks, IP addresses are used (both Home Address and Care-of Address), SIP URIs are used in SIP applications, MAC addresses are used at L2, and different other identifiers are used in multiple system components. The Virtual Identity (VID) concept allows a binding between all these different identifiers. The architecture will support identifier derivation schemes and identifiers management schemes, in order to manage the new identifier space created. Specifically, mappings between identifiers are essential for the cross-layer internetworking of various systems components.

A simple, but yet general identifier model is proposed. It is used as a reference for the design, specification and implementation of interactions between entities and modules which are part of the MT and the overall network infrastructure. The model can be briefly represented by a (1-n-m) notation, where ”1” denotes the VID, ”n” denotes the number of terminals in use by the Virtual User and ”m” denotes the total number of network interfaces (NIC cards) configured in the terminal. This model permits that a single VID can be used in multiples terminals, spanning to the various terminals’ configured interfaces. This aspect of configured interfaces is an essential aspect. The usage of a terminal by a user does not imply that all the interfaces will be used; it will depend on its profile and on the contracted network access. Furthermore, multiple VIDs can be using the same terminal at a given time. In this scenario each VID should be managed independently, due to privacy reasons. Figure 4.8 shows a representation of these concepts.

As represented in figure 4.8, the Virtual Interface Proxy (VIP) abstracts the terminal’ physical interfaces into Virtual Interfaces (VIF) and manages the binding between them. VIF are usable on a per VID basis, that is, each VIF is bind to the corresponding VID for whom it was created. The VIP also provides the Virtual MAC infrastructure, based on the VIF paradigm, which enhances the user’s privacy at the Link Layer. This enhances privacy, since the Virtual MAC (VMAC) paradigm emulates two different devices from the network’s point of view. The VMAC will offer to each VIF a different MAC address. Combining the Virtual Interfaces with the Virtual MAC implies the generation of a VMAC for each VIF.

The usage of Virtual MACs makes the identifier model (1-n-m) dynamic. For this new dynamic model, a flexible binding is required. The binding between the identity and the identifiers can happen at several levels. For example, using typical identifiers, binding can occur in:

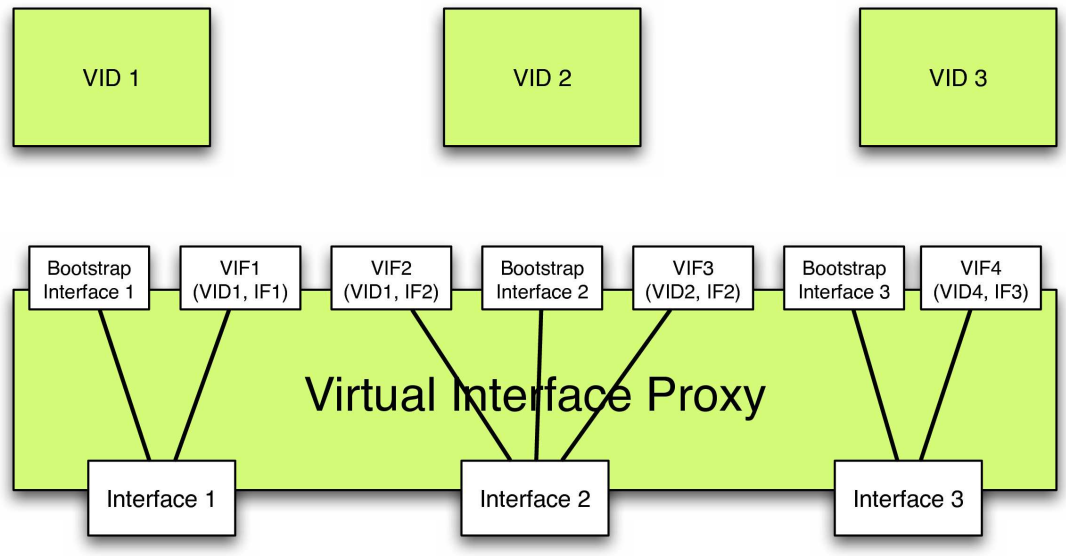

Figure 4.8: Virtual Interfaces

1. A single user (physical person) is identified in the system with one or more VIDs. A VID identifier is a permanent representation (is one-to-many relation; permanent).

2. A single MN, under the usage of a specific VID owner, is identified in the system with a single Home Address (HoA). HoA is a permanent identifier (one-to-one relation; permanent). Different MNs use different HoA. A MN is identified with the HoA and it remains constant until deregistration with that MN. This binding is managed at the Home Agent level.

3. A single network interface (NIC) of a MN is identified with one or more topologically correct care-of addresses (CoA). The CoA is transient, since it is designed to reflect the current point of attachment to the network (one-to-many; transient). Different NICs use different CoA. A single HoA can be associated by the system to one or more CoAs. This binding is managed at the mobility domain, during the Mobile IP registration procedure of a given CoA.

4. Other identifiers are also used for network interfaces, such as, e.g. virtual link local addresses. These are dynamically set by the MN according to the VID using it.

These binding or mapping between one identity and one identifier are managed at different levels and by different entities. Any change in the transient bindings must be conducted under the control of the appropriate system component and then made available to all authorized components that may need to use the binding.

4.5 MANET Mobility Architecture

The solution for the MANET mobility architecture is presented next and is divided in the mobile node and gateway architecture, where all the modules that need to be in place are represented. The solution is also composed by the mobility signaling diagrams, that explains how the mobility is executed and how the modules interact between them to perform the handover.

4.5.1 Mobile Node and Gateway architecture

The MANET general architecture is divided in two main logic units, the Mobile node and the Access Router or Gateway. These two functional entities are very similar in the way they operate, but still they have significant differences. The Access Router contains the Local Mobility Protocol, and thus is also the MAG; the Mobile Node contains the Virtual Interface Proxy (VIP) for managing the virtual interfaces and providing support for the VID paradigm. The MANET Modules have to be designed taking into account this little difference – Not having the Virtual Interface Proxy in the AR.

4.5.1.1 Mobile Node Architecture

The mobile node architecture is represented in figure 4.9. This architecture is derived from the Daidalos project main mobile node architecture.

For space limitations and to keep the diagram clean, figure 4.10 shows the contents of the MT_MANET_MODULES box show in figure 4.9.

The modules that implement the basic mobile node functionalities are also present in figure 4.9. The IEEE 802.21 MIH function and virtual interface proxy (VIP) are easily identifiable. Also represented are the drivers and driver controllers (RALs) of other technologies, in order to show that the MN belongs to heterogeneous environment. The MT_MTC represents the mobile terminal controller (MTC), an entity that controls the most sensible parts of the MN management architecture. Inside MTC are the Virtual Identity Manager (VID_Manager) and the Intelligent Interface Selector (IIS). The MTC also interfaces with upper management modules, such as User preferences modules, control modules and context aware modules. Combining the Upper layers management modules and the Virtual Identities information, the IIS can chose the best interface that each traffic flow of the terminal should use. Also, IIS

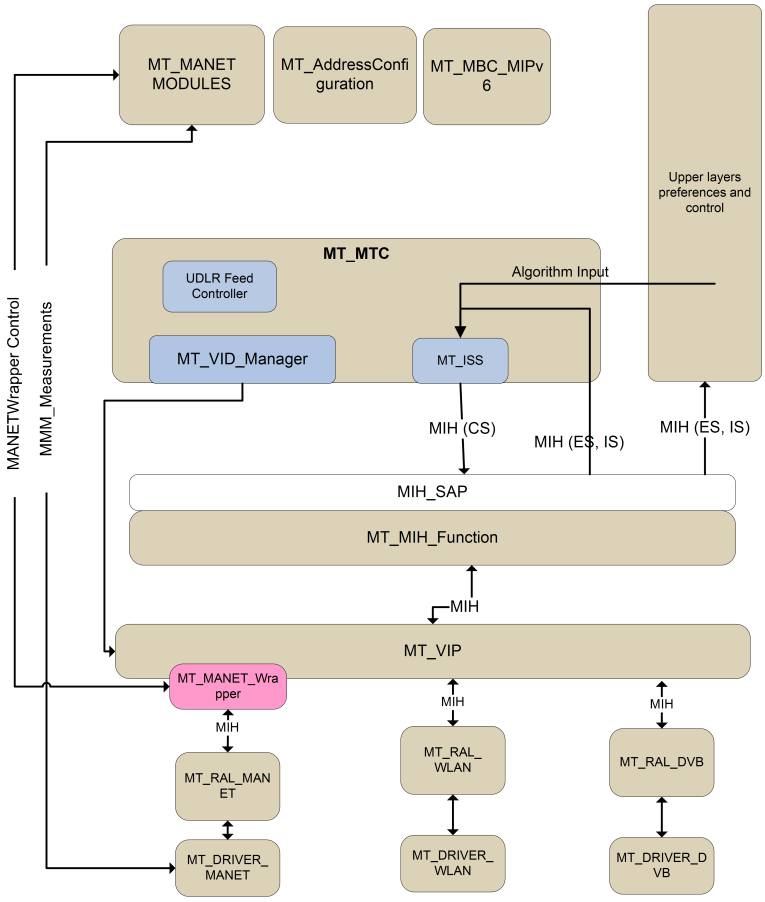

Figure 4.9: Mobile Node architecture

decides to where and when a certain flow or interface should perform handover. The modules responsible for handling multihoming and global mobility management (MT_MBC_MIPv6) are also represented

All the necessary modules to provide a proper multiservice ad hoc network, similar to the one evaluated in Chapter 3 (but upgraded to the new architecture) are present in figure 4.10. The routing protocols AODV (MT_AODV), OLSR (MT_OLSR) and MMARP (MT_MMARP) are all connected to the Routing Manager (MT_RM). RM is the module responsible for maintaining the ad hoc related routing tables (common to all routing protocols) and also for control and select the correct routing protocol according to the VID preferences and the current network. The auto-configuration module (MT_MAGINFO) is the module responsible for providing the MANET with auto-configuration capabilities, also participating in the process of handover candidates discovery, with the validation of the networks found. In the right side of the picture are the QoS related modules. QoS is provided with the modified version of the SWAN protocol discussed in the previous Chapter 3 (section 3.1.4). The

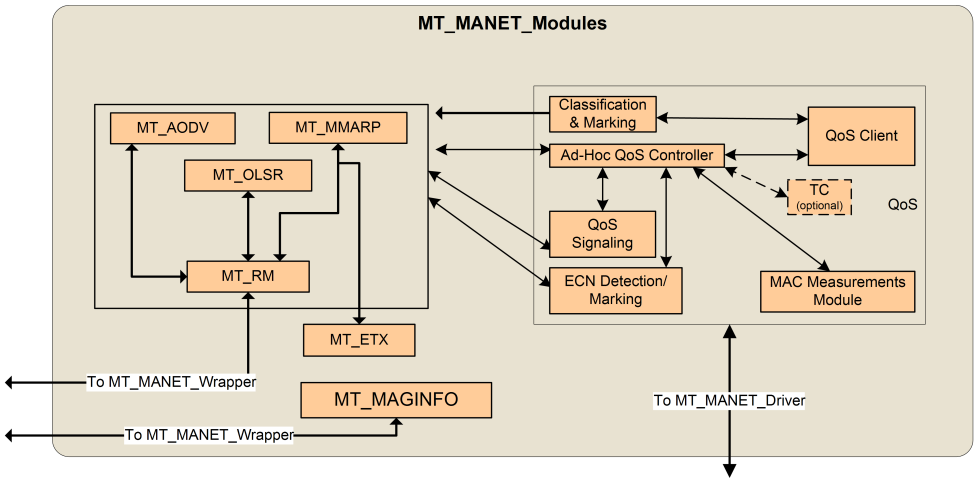

Figure 4.10: MANET mobile node modules in detail

only difference is that the wireless hardware used is QoS capable (IEEE 802.11e), making it possible to implement the traffic queues in hardware rather than by a software daemon.

4.5.1.2 Gateway Architecture

The MANET gateway (GW) is a derivation of the infrastructure Access Router, the only difference is the addition of the MANET modules. The MANET gateway is also the Access Router of the ad hoc network and MAG. The ad hoc modules present in the GW are the same modules used in the mobile node, but without the need to support multiple identities. In addition, the MANET Wrapper at the GW can be much simpler, because it does not need to support multihoming, only functioning as a IEEE 802.21 wrapper. Figure 4.11 shows the gateway architecture.

In the GW it is found the Local Mobility Protocol Engine (AR_LMP_Engine) and the Load Balancing Module (AR_LBM). The Access Router can also manage several networks at the same time, thus the several drivers and drivers controllers shown in the architecture. It is worth noticing the absence of the Virtual Interface Proxy module in the AR simplifying its operation.

The detailed architecture of the MANET modules at the gateway is present in figure 4.12. The differences between the gateway and the mobile node are almost none, except for the multicast routing modules, which, as explained in section 3.2.3 do not need especial MANET modules, using regular multicast routing already present in the gateway.

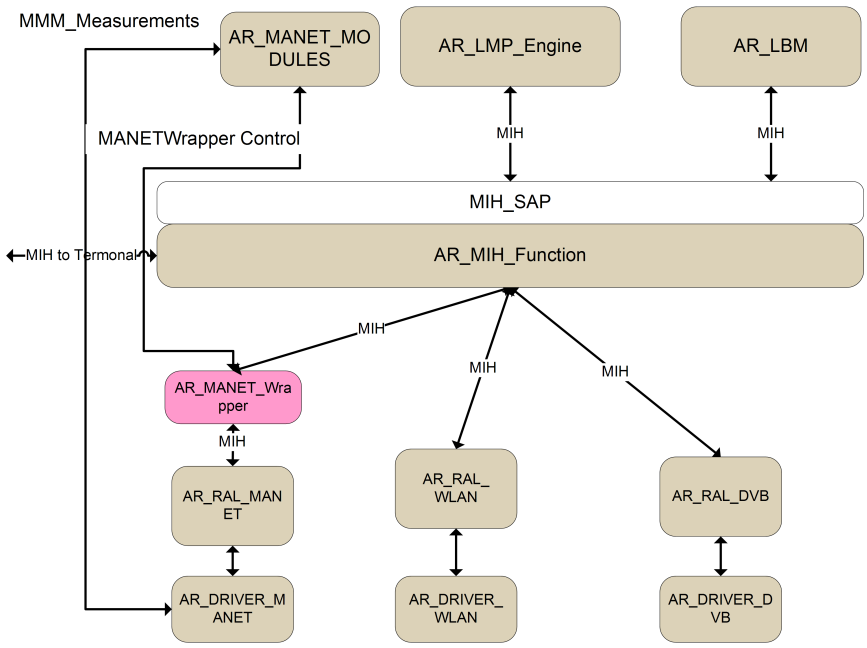

Figure 4.11: Gateway architecture

4.5.2 Mobility Execution and signaling

The general mobility architecture (see section 4.3) was designed to support all kinds of handover. The handover can be defined by who initiates the handover, since both network initiated handover (NIHO) and mobile node initiated handover (MIHO) are supported. It also can be differentiated by when the handover is triggered, if it is after the current link breakage, making it a make-before-brake soft handover, of after the link breakage, occasioning a break-before-make hard handover.

The handover process does not only differentiates by when it is prepared or by whom starts the process, is is also differentiated by what is handed over. The general mobility architecture and mobility process was designed to not only support traditional mobile node handover from one access network to another, but also handover of flows from one interface to another, when the MN is connected to several different networks at the same time. In addition, since the mobile node supports the virtual identity paradigm (see section 4.4.5) that makes each VID appear to network as one distinct and independent MN, all of the handover types are supported inside one VID, independently of another. The only thing not supported, by design, is the handover of flows between VID.

All these types and methods of handover gives to the architecture a good level of flexibility regarding to node movement, hopefully giving the final user a great experience when

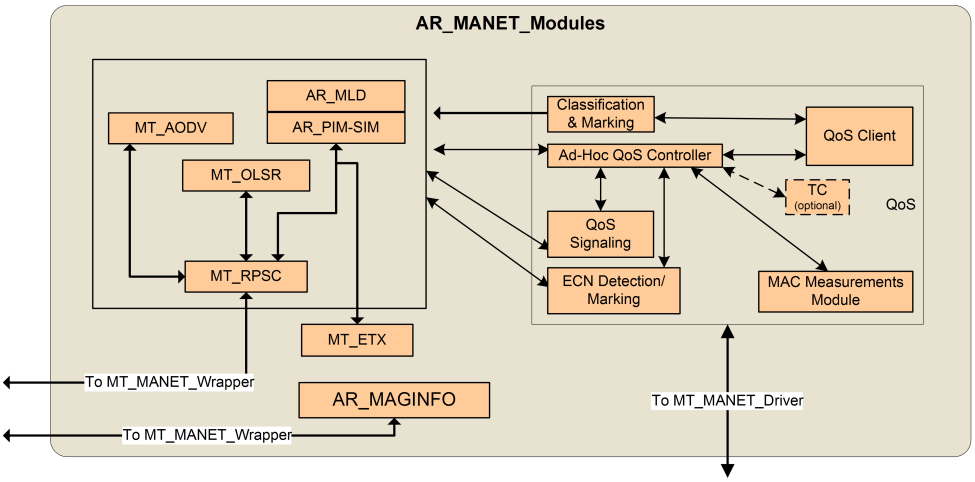

Figure 4.12: Gateway MANET modules in detail

roaming between networks. The diagrams shown and described in this section refer to a MIHO in a make-before-brake situation.

4.5.2.1 Modules involved in MANET mobility

In a way or another all modules are involved in the handover process. However, only a part of them are involved in the realization of handover or in the handover preparation. One of the most important modules for this task is the MANET Wrapper, since is this module that provides the necessary ”glue” allowing that a MANET node can participate in a IEEE 802.21 based network.

MANET Wrapper has 3 important tasks related to mobility preparation and execution:

- Performs an abstraction of the MANET to the MIH Function, presenting it as a new technology, different of regular IEEE 802.11 but similar.
- Collects events (MIH Event Service) from the RAL_WLAN and Information from both the auto-configuration module and the routing protocols. It processes this information and then produces events to the MIHF, that have relevance in the ad hoc context.
- Controls the process of handover candidates validation, coordinating the driver and auto-configuration protocol.

To provide all these functionalities, the MANET Wrapper (MW) has interfaces to with the Media Independent Function (MIHF), with the RAL 802.11 (RAL_WLAN), the auto-

configuration module (MAGINFO) and with the routing protocol manager (RM). From the MIHF/VIP point of view, the MW acts like a RAL, since it implements the MIH-LINK-SAP (MIH link Service Access Point). A MIH-LINK-SAP is also implemented towards the RAL_WLAN, making it appear like a real MIHF. This way the MW becomes transparent to both the MIHF and the RAL. The interaction with the RM is only to control the behavior of routing protocol during handover (activating and deactivating them) and also to enable the routing protocols to contribute with relevant information that can help to detect networks breaks of connectivity. From MAGINFO, the MW will obtain information about the available MANETs and Gateways; information which includes the necessary metrics that characterize the path to each available gateway.

4.5.2.2 Candidates discovery signaling

Before the handover execution, the MN will have to search for suitable candidates. This search can happen whether the node is on ad hoc mode or not. For example, the node can be using a 3GPP network and decides to scan all available interfaces for another (better, cheaper, etc.) type of networks access, this will trigger an ad-hoc network scan for handover candidates.

After the MN has decided that it needs to scan for networks, the MIHF will receive a command ordering a scan all interfaces and report results (MIH_ScanRequest). The MIHF will then issue a scan command (Link_ScanRequest) for each available interface in the MN. When the VIP forwards the request to the Ad hoc interface, the MANET Wrapper will intercept the request an control the ad hoc scan from there on. First, it will order a standard scan to the driver and receive the list of networks. Then, it will connect to each one of the networks found and request the auto-configuration (MAGINFO) module to obtain more details about the network. When the auto-configuration module has obtained such details, it transmits them to the MW, who will in turn generate a scan response message, providing not only the standard information but also the ad hoc specific information too. These new results, enriched with the ad hoc extra information will then be forwarded to the MIHF in response the original scan request (MIH_Scan) by the intelligence in the mobile node (Intelligent Interface Selector – IIS). Figure 4.13 shows the internal mobile node messages for handover candidates discovery described above.

After the IIS has received the results for all the scans requested, it can take decisions about what interfaces should be used, and what flows should be assigned to each interface.

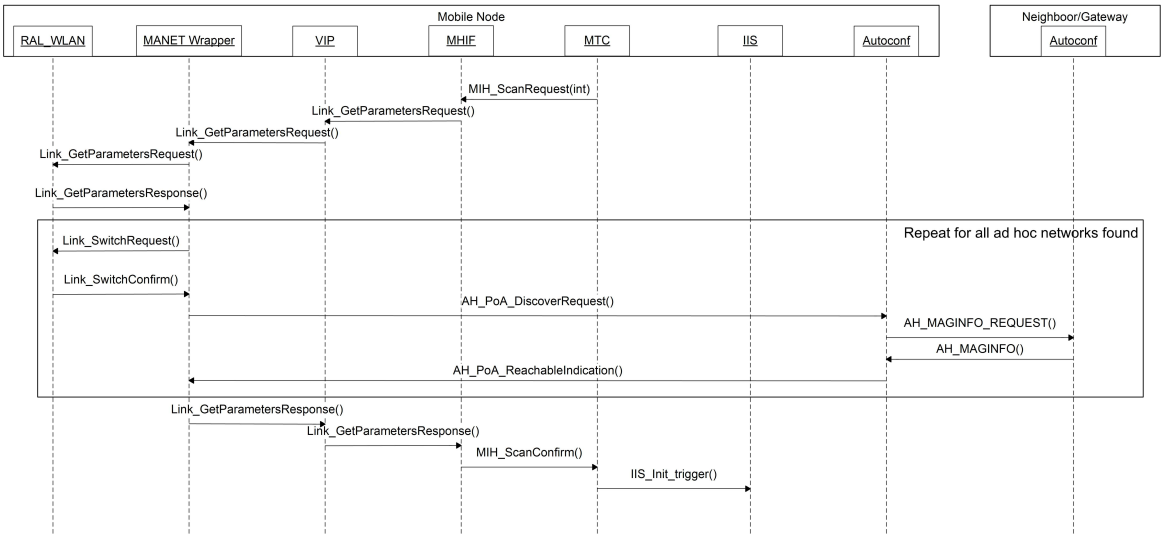

Figure 4.13: Handover candidates discovery, simple MSC

4.5.2.3 Handover Signaling

The MANET was developed in order to fully integrate in the general mobility architecture (see previous section). Therefore, it will use the same messages for mobility signaling purposes. These messages are defined in the IEEE 802.21 MIH Standard. From the point of view of the MANET there are only two situations of interest; the handover is from any other technology with destination in the MANET, or the handover is started when the node is on the MANET and moves to any other technology. Since the mobile node uses the MIHF, it is not important from where the node comes, or goes, only the MIHF and the related technologies have to know the details.

As stated above, the mobility signaling is performed recurring to the IEEE 802.21 messages. Figure 4.14 shows the external signaling between the mobile node and the networks elements during a handover from 802.11 WLAN to a MANET.

The Handover process can be separated in three phases: handover preparation, handover execution and handover conclusion. The handover preparation phase starts when the IIS sends the order to perform handover (from interface A to interface B) to the MIHF, which will coordinate the rest of the handover process. The MN starts the process by communicating to the core network its intentions to perform handover, using the Handover Initiate message (HO_Initiate.req). This message contains information about the MN, the destination target for handover (destination AR), and QoS related information. The LMD (by means of the LMA, the Policy Decision Point and Access Routers) analyses the handover request to

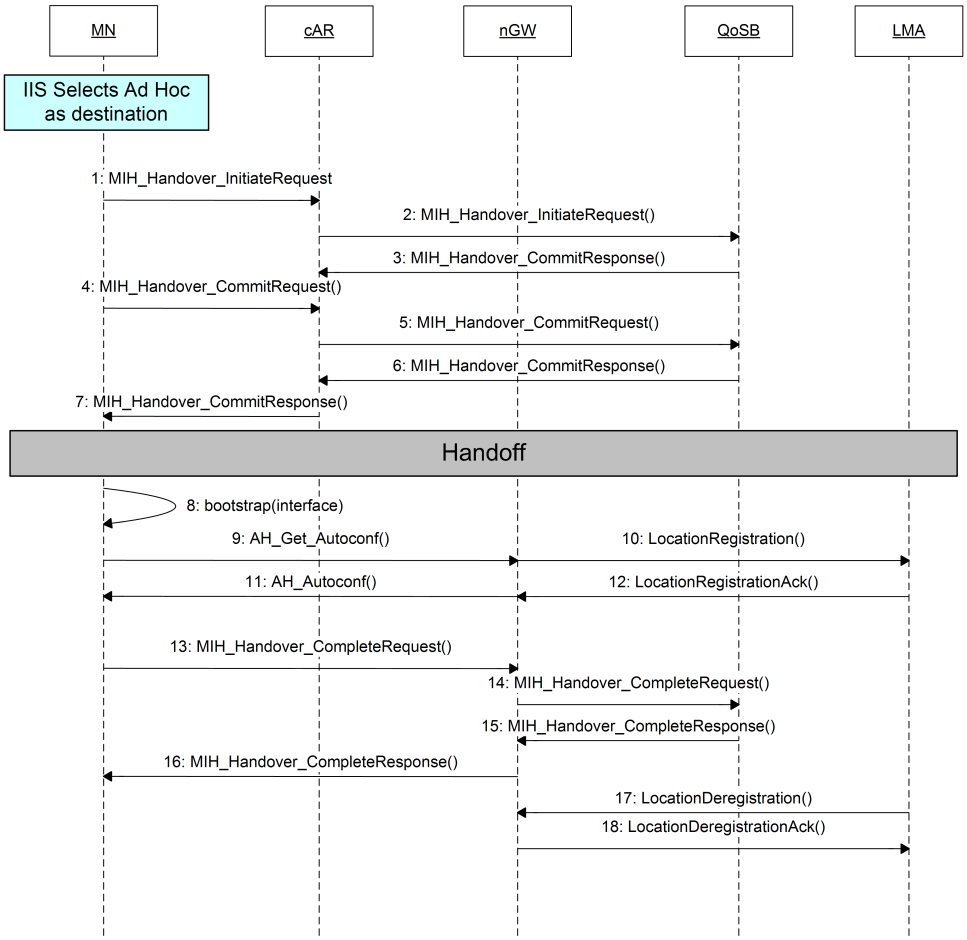

Figure 4.14: Handover signaling

determine if the handover can be effectuated. In case the core and destination access network have the necessary resources to accommodate the new terminal/session, the necessary network resources are allocated and the MN receives a response indicating that it can perform the handover (HO_Inititate.resp).

After proper authorization, the second phase of the handover process (execution phase) begins. The MN sends a message to the network (HO_Commit.req) to indicate it is performing handover and that the resources need to be committed (really reserved); after receiving confirmation of the resources commitment (HO_Commit.resp), the handoff to the new network is performed. At this time the Ad Hoc interface must be bootstrapped and fully configured. During the interface setup, the new AR (GW) detects the presence of the MN in its access network and triggers the Local Mobility Protocol to update the new location of the MN (LocationRegistration messages).

The third and final phase of the handover process is the conclusion phase. It is started when the ad hoc interface finishes the bootstrap. The MN sends a complete handover request (HO_CompleteResquest) to the network, so that all network elements involved in the handover process can acknowledge a successful handover. During this process, the LMA also de-registers the location of the MN still present on the old Access Router. The network responds with a complete handover confirmation to the MN, concluding that way the handover process.

4.5.2.4 Internal Signaling during Handover

The signaling presented in the previous section is not the only relevant signaling occurring during the handover process. The Mobile Node is composed by several different modules (see section 4.5.1.1) that exchange messages internally in order to coordinate the handover process. Figure 4.15 shows the MSC of the internal signaling for the same handover described in the previous section (MIHO from infrastructure to ad-hoc).

Not long after the conclusion of a *handover candidate discovery* the MTC provides the IIS with the scan results, the IIS then decides to perform a handover. The handover starts when the IIS gives to the MTC the map of flows and interfaces for each VID; the MTC compares with the current map and if there are differences, it initiates the handover to comply with the new allocation map.

The MTC starts the signaling when it sends a Handover_InititateRequest to the MIHF, which will then send to the Network. The MN then exchanges the other 802.21 messages until the Handover_CommitResponse arrives to the MIHF from the network. At this time the MIHF sends a SwitchRequest to the 802.11 driver controller (RAL) causing the L2 handoff (change from managed mode to ad hoc mode). Until this point the messages exchanged are common to all handovers performed by the MN and to all technologies supported.

After the 802.11 driver changes to ad hoc, the Manet Wrapper starts its operation. First it activates the Auto-configuration module, which immediately requests to the network an AH_AUTOCONF message. The MW also starts the Routing protocol (AODV in this case), so that multihop messages can be sent and correctly routed. Only when both the Auto-configuration and routing protocol have acknowledge the activation, is the switch confirmation (SwitchtConfirm) forwarded by the MW to the MTC. By this time the MN has performed the handover and is ready to integrate in the network.

The final step, once again common to all handover processes (MIHO), is the conclusion

of the handover, done by the MTC and MIHF with the exchange of the handover complete (Handover_Complete Request and Response) messages with the network.

4.6 Issues in MANET Architecture

The mobility architecture described in the previous section allows the integration of MANET in an architecture designed, primary, to support single hop access networks. One of the key elements in the design that permits this seamless integration is the MANET Wrapper module. The location of the MANET Wrapper, just above the 802.11 driver, permits that the MANET appears as a regular, single hop network, to the Media Independent Handover function in the terminal, while at the same time, enables the correct functioning of the MANET. However, this solution only provides a clean integration with the IEEE 802.21 MIH standard.

Another key module in the architecture is the Virtual Interface Proxy (see 4.4.5), which allows the virtualization of the mobile node physical interfaces. This module will then create a set of interfaces for each VID, appearing as if each VID is a different and independent mobile node.

This design of the MANET architecture, focused on the compatibility with the IEEE 802.21 MIH standard, with support for virtual identities and with support for multihoming has some issues in normal ad hoc operation. These issues, and others, will be discussed in the next sections.

4.6.1 Multiple routing protocols

The architecture presented in section 4.5.1 has support for two routing protocols: AODV and OLSR are present in both the gateway and the mobile node. The inclusion of two routing protocols was performed to give the operators more choices regarding what routing protocol to use according to the type of access network deployed. In addition, operators can implement a mechanism that changes the routing protocol dynamically according to network conditions.

This scenario can lead to an access network that is running simultaneously both routing protocols (with one or more gateways), which is a situation that prevents the efficient operation of the MANET. Consider the following scenario for a better explanation of the issue.

An operator has one MANET access network in the LMD, and its gateway has the two routing protocols running, in order to provide better compatibility. Currently in the network are 6 nodes, arranged in a string topology. From the gateway to the 4th node, the OLSR protocol is in use and the network operates properly. However, the 5th node, which has just made roaming from another operator network, only can run AODV protocol, due to policy restriction, for instance. In this situation, only the four nodes closest to the gateway have service, although all the other nodes, except from node 5 have OLSR available. This scenario could be prevented if only one routing protocol is allowed for all nodes.

Another example, that proves inefficiency, is when in a network where the gateway has the two routing protocols running, but where then nodes have only either of then. In this scenario, instead of only one, more robust and cooperating network, there will be two overlay networks, one running AODV and other running OLSR, that can only communicate with each other through the gateway, even if the actual peers are link layer neighbors. In addition, in this scenario, nodes can get easily get isolated after movement, when they reach an area that does not have 'routing protocol coverage'.

The fact that the architecture is designed with support for both routing protocols, is not synonym that all mobile nodes will have both. For instance, an operator can have different service plans, and only the more expensive supports both protocols simultaneously, whether other service plans support only routing protocol. Handover choices may also get affected, since the target network can not have (at least close to the mobile node location) the routing protocol needed to support the mobile node.

The cleanest way to solve these problems is to enforce all nodes to have both routing protocols available regardless of the service plan, and also, only permit that only a routing protocol is used per gateway. This simple measures will mitigate, if not prevent, the problems presented.

4.6.2 Virtual interfaces

The implementation of the Virtual Identity paradigm is done through virtual interfaces. Each VID active in the terminal will have a set of virtual interfaces, that map to real interfaces, in the mobile node. The virtual interfaces of one VID are isolated from other VID virtual interfaces, providing this way the desired privacy for all the VID running in the mobile node.

The virtual interfaces area also present for ad hoc mobile nodes, which may introduce some problems. One of the problems relates to the routing protocol issues described in the

previous section (4.6.1). Since a VID is defined by a set of rules and policies, even if the node has both routing protocols available, a VID can only use one of them. This behavior increases the routing protocols issues previously described.

Another problem created by the virtualization of interfaces is related the behavior of the software responsible for implementing the routing protocol functionality. In the presence of several virtual nodes (one for each active VID), the most natural solution is that each VID has an instance of the routing protocol running. However, this solution is not optimal, since it will create more network overhead for each mobile node (which will be seen as several nodes by the network), and also, upon a routing protocol message reception, it is not clear to which VID will be delivered. For example when the network is running AODV and a neighbor sends a RREQ (Route Request) message, it is not clear which instance of the AODV software running in the mobile node will respond (and possibly forward) to the request. However, two routing protocols can be running at the same time, provided that one VID runs one (e.g AODV) and other VID runs the other (e.g. OLSR). This scenario is possible but limits the number of VID using the MANET to the number of available routing protocols. One routing protocol can only be shared by more than one VID when the signaling messages' reception problem is solved.

Another possibility is that only one instance of the routing protocol is running at each time for all VIDs. This solution eliminates the ambiguity present if multiples instances are running, but has other problems: for start, it will add more complexity to the software, since it will have to manage several interfaces, each one with one different address; and it will need that the routing protocol will be identity aware, bringing possible privacy issues.

The second option appears to be better, since it is more easy to implement software to manage several interfaces than to cope with the ambiguity when receiving multiple messages.

The problems presented and exemplified here for the routing protocols exist also for all other protocols. From auto-configuration to QoS, passing by multicast, all software will suffer with this virtualization of interfaces.

4.6.3 Multiple gateways and multihoming

Multihoming at the Global Mobility Domain requires the extension of Mobile IPv6 so that it supports multiple CoA registration. The registration is performed using Binding Update messages either to the Home Agent or the Correspondent Node in case of routing optimization. In order to handle distribution of flows based on policies, each flow has to be identified

by a 5-uple (IP source address, IP destination address, source port, destination port and protocol type). This 5-uple is identified by a Flow ID (FID). FID have to be communicated to the Home Agent or Correspondent Node. FID are then bound with CoAs. Using this scheme, each time a packet arrives to the Home Agent marked with certain FID, the HA can chose the right CoA as the destination. This extension support Mobile Nodes in a MANET with multiple gateways (each gateway in a different LMD).

When the multiple gateways are present in the same LMD, multihoming has to be executed at the Local Mobility Domain Level. In the same LMD, multihoming is offered in a per-gateway basis, that is, for each gateway reachable by the mobile node, a new virtual interface is created (for each VID). The multihoming aware module (MT_LBM) will the be responsible for taking advantage of the multihoming created.

In a network with two gateways, each VID will have, apart from the normal ad hoc interface, one virtual interface for each gateway. Each of this virtual interface represents a tunnel (IPv6-over-IPv6) created between the interface and the gateway. When traffic is sent through one of these interfaces, the MANET Wrapper will encapsulate it and send it directly to the respective gateway, which de-encapsulates the traffic and sends it to its destination.

The presence of the interface, not bound to a tunnel, is required to provide better network operation. In the first place, traffic addressed to local nodes (neighbors) do not need to go through the gateway and then back to the destination mobile node. Also, this 'original' interface is useful so that traffic not needing multihoming features, can travel in the air without the tunnel overhead.

Traffic reception is other issue present in multihoming at the LMD level. When a node has two or more gateways available in the same LMD, the virtual interfaces created for multihoming will all have the same Care-of Address. This situation brings the issue of how a correspondent node sends traffic to the mobile node. To solve this issue the LMD has to be multihoming aware. When a mobile node creates a virtual interface for multihoming, it has to mark the traffic that uses the new interface with a flow identifier (FID), and has to communicate this FID to the gateway. The gateway then will communicate the FID, along with the associated CoA and the gateway identifier to the LMA. The LMA will construct a tree-way routing table, so that traffic coming to the local domain, destined to a certain CoA and marked marked with one FID can be directed to the right gateway.

4.7 Conclusion

In this chapter the traditional solutions for performing mobility in IP-based networks were presented, along with the motivation for supporting mobility. New mobility paradigms, such as the in development standard IEEE 802.21 Media Independent Handover and the Localized Mobility Management concept being developed inside the IETF, were also presented and described. The DAIDALOS general mobility architecture, which combines in a seamless way, both traditional and new mobility management paradigms was introduced and explained. This architecture integrates several different access technologies, making the network ubiquitous to the user, and provides the ability for a node to handover from one access technology to another in a seamless way.

After the proper technologies were introduced, the MANET adaptations needed to integrate in the general mobility architecture were introduced. THe underlining problems that appear when a MANET is integrated with a network designed to support only single-hop access technologies were identified. Then each of the presented problems were further analyzed and a solution presented. The presented solutions were developed with the objective of maintaining the core network MANET unaware, which originated the MANET to be molded and changed according to the requirements.

The final MANET architecture supports all the features of the general architecture. a MANET mobile node will be able to use the MANET in the same way as it uses infrastructure networks, suffering only of a performance decrease. When a better network is found by the mobile node, it can seamlessly handover to that better network (completely or partially).

Although the developed architecture is well integrated in the general architecture, there are some issues that still need a proper solution. These issues were identified and a compromise solution was presented for each one. Further research will be performed so that these issues can have a solution more adequate to the desired functionality and not limiting the functioning of the network.

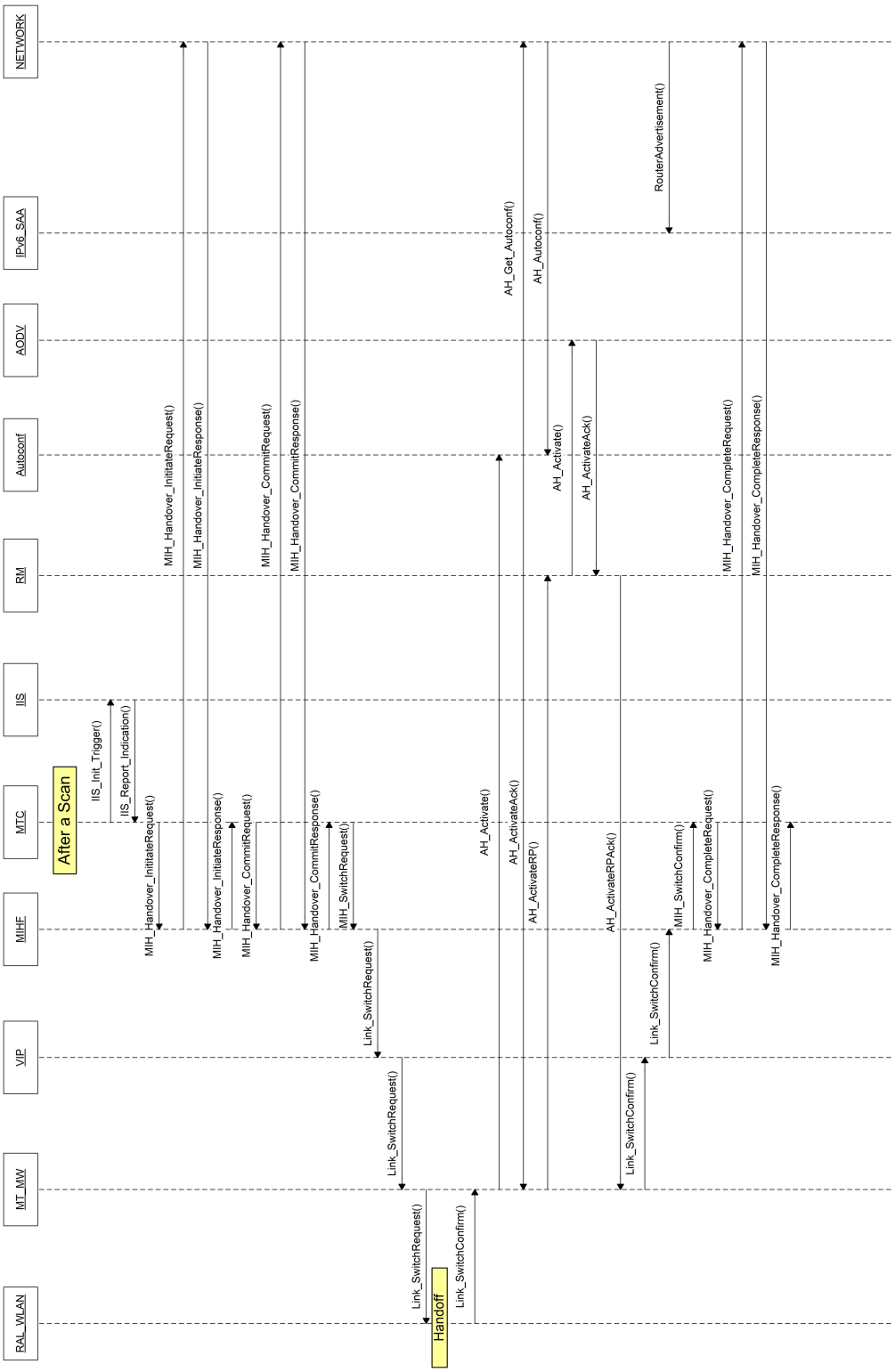

Figure 4.15: Handover, Internal mobile node signalling

Chapter 5

Conclusions

In general, self-organization in the context of networks is a very attractive concept. Network solutions that can automatically configure and organize in order to provide the best solution and performance possible will be of much interest, not only for operators and service providers, but also for regular users. The leading research towards these self-organized networks is the research performed in the various types of ad hoc networks. Current proposals for ad hoc networks already provide good solutions for individual problems, but there are still work to do regarding integration of the various solutions.

The testbed presented in this thesis, and developed as a demonstrator for the integration of mobile ad hoc networks as hotspot networks extensions, have proven that it can be possible to integrate an ad hoc network, making it part of the infrastructure, and offering the full set of services needed for a good integration. The tests were envisioned to be conducted in an incremental way, by adding one functionality on top of another, previously evaluated functionality, was easy to determine the functionalities that were introducing the most penalties in the network operation.

The obtained results from the complete evaluation have shown that the resulting ad hoc network has modest performance. The maximum throughput achieved is just a sample of the networks performance. The obtained results shown that the available throughput decreases almost by a factor of two, for every hop that is added to the network, thus, imposing a limit in the number of hops that the network can have. The added protocol overhead, needed to support the required services, may be too much for the gained services.

Assembling a testbed with so many different protocols, computers and other software has also proven to be a complex task. In addition, the many interactions between the protocols

makes the complete system very prone to errors and crashes. This experience showed that, in order for an integrated solution to be resilient to all kinds of users and usage, the quality of the software has to be very good.

The integrated testbed developed has almost all the pre-established requirements covered, except for user mobility from and to the ad hoc network. Therefore, there is still work to be done regarding that problem. The developed tested was part of the phase one of the DAIDALOS project. The second phase if this project, kickoff in January 2006, has inherited the mobility problem. The overall mobility architecture of the DAIDALOS project was updated and so it was the ad hoc architecture, in order to integrate whit the new mobility architecture.

New protocols and paradigms were also included in the new architecture. Among then, the IEEE 802.21 Media Independent Handover, and the network-based localized mobility management are some of the most important. Localized mobility management is a good added value to the network. It separates the mobility management in local and global domain, promising to reduce handover times at the same time that reduces the network overhead caused by the mobility support. The IEEE 802.21 protocol allows that the traditional mobility protocols (such as mobile IP) do not have to know the specifics of each technology used at the terminal. However, the inclusion of this protocol has conditioned the design of the architecture of the MANET in a way that the solution is completely integrated and dependent of the presence of IEEE 802.21 in the network.

The mobility architecture developed allows the integration of MANET in an architecture designed, primary, to support single hop access networks. The introduction of the MANET Wrapper, just above the 802.11 driver, permits that the MANET appears as a regular, single hop network, to the Media Independent Handover Function in the terminal, while at the same time, permits that the MANET continues functioning normally. However, this approach can have performance penalties in the response time to link breakage events.

The MANET Wrapper has also a key role in the way multihoming is handled at the local mobility domain level. When a mobile node can reach more than one gateway in the same network, the MANET Wrapper is responsible, in conjunction with other modules in the terminal, for the creation of the virtual interface for each reachable gateway. The multihoming aware modules in the mobile node will then use these new virtual interfaces to perform the multihoming and load balancing.

This mobility architecture still contains some issues. Many things can cease to work

properly if no care is taken during the implementation of the solution. Among the most problematic modules are the routing protocols. The support for two routing protocols in the network increases the challenges in the ad hoc network architecture. Future developments and research is still needed to clearly address these challenges.

The near-term future work consist in implementing a new testbed for the demonstration of the newly engineered architecture. With this new testbed operational, the tests performed in the existing testbed should be repeated, so that both solutions can be compared, from the point of view of the evolution achieved.

Long term future work will be focused on solving the issues presented in section 4.6. Especially, the case of multiple virtual identities running the same routing protocol, without causing problems in the message reception, will be one of the key research topics. Related to this issue is the possibility of two different routing protocols sharing the same interface. This scenario will permit that an ad hoc network with only one gateway can have both routing protocols active.

Bibliography

Publications

[1] Hakim Badis and Khaldoun Agha. QOLSR, QoS routing for ad hoc networks using OLSR. *European Transactions on Telecommunications*, 16(5):427–442, 20005.

[2] João Paulo Barraca, Susana Sargento, and Rui L. Aguiar. Polynomial-assisted ad hoc charging protocol. *10th IEEE Symposium on Computers and Communications (ISCC)*, pages 945–952, 2005.

[3] Elizabeth M. Belding-royer. Report on the AODV Interop. *UCSB Tech Report*, 2002-16, 2002.

[4] N Ben Salem, L Buttyan, J. P. Hubaux, and M Jakobsson. Node Cooperation in Hybrid Ad Hoc Networks. *IEEE Transactions on Mobile Computing*, 5(4):365–376, April 2006.

[5] Tânia Calçada and Manuel Ricardo. Extending the coverage of a 4G Telecom Network using Hybrid Ad-hoc Networks: a Case of Study. In *4th Annual Mediterranean Ad Hoc Networking Workshop*, June 2005.

[6] D Clark, R Braden, A Falk, and Pingali V. FARA: Reorganizing the Addressing Architecture. FDNA Workshop, ACM SIGCOMM 2003, August 2003.

[7] T Clausen, Philippe Jacket, A Laouiti, P Muhlethaler, A Qayyum, and L Viennot. Optimized Link State Routing Protocol. IEEE International Multitopic Conference, INMIC2001, 2001.

[8] Sergio Crisóstomo, Susana Sargento, Marek Natkaniec, and Norbert Vicari. A QoS Architecture Integrating Mobile Ad hoc and infrastructure Networks. In *3rd ACS/IEEE*

International Conference on Computer Systems and Applications (AICCSA), January 2005.

[9] Ahn Gahng-Seop, Andrew T. Campbell, Andras Veres, and Li-Hsiang Sun. Supporting Service Differentiation for Real Time and Best-effort Traffic in Stateless Wireless Ad Hoc. *IEEE Transactions on Mobile Computing*, 1(3):192–207, 2002.

[10] Francisco Galera, Pedro Ruiz, and Antonio Gomez-Skarmeta. Security Extensions to MMARP Through Cryptographically Generated Addresses. To appear in Lecture Notes in Informatics.

[11] João Girão, João Paulo Barraca, Bernd Lamparter, Dirk Whesthoff, and Rui L Aguiar. QoS-differentiated Secure Charging in Ad-hoc Environments. International Conference on Telecommunications - ICT2004, 2004.

[12] Christophe Jelger and Christian Tschudin. Dynamic Names and Private Address Maps: Complete Self-Configuration for MANETs. 2nd Conference on Future Networking Technologies (CoNEXT'06), 2006.

[13] Bernd Lamparter, João Girão, Dirk Whesthoff, Rui L Aguiar, and João Paulo Barraca. Linking Ad-hoc Charging Schemes to AAAC Architectures. Security in Ad-hoc and Sensor Networks: First European Workshop ESAS 2004.

[14] Bernd Lamparter, K Paul, and Dirk Whesthoff. Charging Support for Ad hoc Stub Networks. *Journal of Computer Communication, Special Issue on Internet Pricing and Charging: Algorithms, Technology and Application*, 2003.

[15] Bernd Lamparter, Krishna Paul, and Dirk Westhoff. Charging support for Ad hoc Stub Networks. *Journal of Computer Communications, Special issue on Internet Pricing and charging: Algorithms Technology and Applications*, 26(13):1504 – 1514, 2003.

[16] Seoung-Bum Lee, Gahng-Seop Ahn, Xiaowei Zhang, and Andrew T. Campbell. INSIGNIA: An IP-Based Quality of Service Framework for Mobile ad Hoc Networks. *Journal of Parallel and Distributed Computing*, 60(4):374–406, April 2000.

[17] Pedro Ruiz, Antonio Gomez-Skarmeta, and Ian Grovez. The MMARP Protocol for Efficient Support of Standard IP Multicast Communications in Mobile Ad hoc Access Networks. In *IST Mobile & Wireless Communications Summit*, pages 478–482, June 2003.

[18] Susana Sargento, Tânia Caçada, João Paulo Barraca, Sergio Crisóstomo, João Girão, Marek Natkaniec, Norbert Vican, Francisco Cuesta, and Manoel Ricardo. Mobile Ad-Hoc Networks Integration in the DAIDALOS Architecture. June 2005.

[19] Susana Sargento, Rafael Sarrô, Ricardo Duarte, and Stupar Patrick. Mobility in the Integration of Mobile Ad-hoc Networks. MWCS STREP ENABLE and IP Daidalos Workshop - Research and Deployment Possibilities based on MIPv6, July 2007.

[20] Susana Sargento, Rafael Sarrô, Patrick Supar, Francisco Gallera, Marek Natkaniec, João Paulo Vilela, and João Barros. Ubiquitous Access through the Integration of Mobile Ad hoc Networks. 16th IST Mobile & Wireless Communications Summit, July 2007.

[21] Ion Stoica, Daniel Adkins, Shelley Zhuang, Shenker Scott, and Shonesh Surava. Internet Indirection Infrastructure. ACM SIGCOMM, August 2002.

[22] K. K. Vadde and V. R. Syrotiuk. Quantifying Factors Affecting Quality of Service in Mobile Ad Hoc Networks. *SIMULATION*, 81(8):547–560, 2005.

[23] Sheng Zhong, Jiang Chen, and Yang Richard Yang. SPRITE: A simple, cheat-proof, credit-based system for mobile ad hoc networks. In *INFOCOM 2003. 22nd Annual Conference on the IEEE Computer and Communications Societies. IEEE*, volume 3, pages 1987 – 1997 vol.3, 2003.

RFC and Standards

[24] Cedric Adjih, Saadi Boudjit, Philippe Jacquet, and Anis Laouiti. Address autoconfiguration in Optimized Link State Routing Protocol. IETF Internet draft, `http://tools.ietf.org/id/draft-laouiti-manet-olsr-address-autoconf-01.txt`, July 2005.

[25] Hankim Badin and Khaldoun Agha. Quality of service for Ad hoc Optimized Link State Routing Protocol (QOLSR). IETF Internet draft, `http://www.ietf.org/internet-drafts/draft-badis-manet-qolsr-05.txt`, March 2007.

[26] Fred Baker. An outsider's view of MANET. IETF Internet draft, `http://tools.ietf.org/id/draft-baker-manet-review-01.txt`, March 2002.

[27] Anand Bekedar, Ajoy Singh, Vinod Kumar, and Suresh Kalyanasundaram. A Protocol for Network-based Localized Mobility Management. IETF Internet draft, http://tools.ietf.org/wg/netlmm/draft-singh-netlmm-protocol-02.txt, March 2007.

[28] Jim Bound, Bernie Volz, Ted Lemon, Charlie Perkins, and Mike Carney. Dynamic Host Configuration Protocol for IPv6 (DHCPv6). IETF RFC 3315, http://www.ietf.org/rfc/rfc3315.txt, July 2003.

[29] B Cain, S Deering, I Kouvelas, and B Fenner. Internet Group Management Protocol, version 3. IETF RFC 3376, http://tool.ietf.org/rfc/rfc3376.txt, October 2002.

[30] Ian Chakeres and Charles Perkins. Dynamic MANET On-demand (DYMO) Routing. IETF Internet draft, http://tools.ietf.org/id/draft-ietf-manet-dymo-09.txt, May 2007.

[31] T Clausen and Philippe Jacquet. Optimized Link State Routing Protocol (OLSR). IETF Experimental RCF 3626, http://www.ietf.org/rfc/rfc3626.txt, October 2003.

[32] S Corson and Macker Joe. Mobile Ad hoc Networking (MANET): Routing Protocol Performance Issues and Evaluation Considerations. IETF Informational RFC 2501, http://www.ietf.org/rfc/rfc2501.txt, January 1999.

[33] Sri Gudavelli, Kent Leung, Vijay Devarapalli, Kuntal Chowdhury, and Basavaraj Patil. Proxy Mobile IPv6. IETF Internet draft, http://tools.ietf.org/id/draft-ietf-netlmm-proxymip6-00.txt, April 2007.

[34] IEEE. IEEE Standard for Wireless LAN Medium Accesss Control (MAC) and Physical Layer (PHY) Specifications, 1999.

[35] IEEE. Draft IEEE Standard for Local and Metropolitan Area Networks: Media Independent Handover Services, 2006.

[36] Philippe Jacquet, P Minet, L Laouiti, T Viennot, and C Clausen. Multicast Optimized Link State Routing. IETF Internet draft, http://tools.ietf.org/id/draft-jacquet-olsr-molsr-00.txt, November 2001.

[37] C Jelger, T Noel, and A. Frey. Gateway and Address auto configuration for IPv6 ad hoc networks. IETF Internet Draft, draft-jelger-manet-gateway-autoconf-v6-02.txt, April 2004.

[38] Christophe Jelger, Thomas Noel, and Arnaud Frey. Gateway and address autoconfiguration for IPv6 adhoc networks. IETF Internet draft, http://tools.ietf.org/id/draft-jelger-manet-gateway-autoconf-v6-02.txt, April 2004.

[39] Jaehoon Jeong, Jungsoo Park, Hyoungjun Kim, and Dongkyun Kim. Ad Hoc IP Address Autoconfiguration. IETF Internet draft, http://tools.ietf.org/id/draft-jeong-adhoc-ip-addr-autoconf-02.txt, February 2004.

[40] Jorjeta Jetcheva and David Johnson. The Adaptive Demand-Driven Multicast Routing Protocol for Mobile Ad-Hoc Networks (ADMR). IETF Internet draft, http://tools.ietf.org/id/draft-ietf-manet-admr-00.txt, July 2001.

[41] D. Johnson, C. Perkins, and J. Arkko. Mobility Support in IPv6. IETF RFC 3775, http://tool.ietf.org/rfc/rfc3775.txt, June 2004.

[42] David B Johnson and David A Maltz. The Dynamic Source Routing Protocol for Mobile Ad hoc Networks (DSR). IETF Experimental RFC 4728, http://tools.ietf.org/rfc/rfc4728.txt, February 2007.

[43] James Kempf, Kent Leung, Phil Roberts, Katsutoshi Nishida, Gerardo Giaretta, and Marco Liebsch. Goals for Network-Based Localized Mobility Management (NETLMM). IETF Informational RFC 4831, http://tools.ietf.org/rfc/rfc4831.txt, April 2007.

[44] James Kempf, Kent Leung, Phil Roberts, Katsutoshi Nishida, Gerardo Giaretta, and Marco Liebsch. Problem Statement for Network-Based Localized Mobility Management (NETLMM). IETF Informational RFC, http://tools.ietf.org/rfc/rfc4830.txt, April 2007.

[45] Juien Langanier and Eggert Lars. Host Identity Protocol (HIP) Rendezvous Extension. IETF Internetf draft, http://www.ietf.org/internet-drafts/draft-ietf-hip-rvs-05.txt, June 2006.

[46] A Laouiti, C Adjih, S Boudjit, and P Jacquet. Address Autoconfiguration in Optimized Link State Routing Protocol. IETF Internet Draft, draft-laouiti-manet-olsr-address-autoconf-01.txt, january 2006.

[47] Robert Moskowitz, Pekka Nikander, Petri Jokela, and Thomas R. Henderson. Host Identity Protocol. IETF Internet draft, `http://www.ietf.org/internet-drafts/draft-ietf-hip-base-08.txt`, June 2007.

[48] R Ogier, Fred Templin, and M Lewis. Topology Dissemination Based on Reverse-Path Forwarding (TBRPF). IETF RFC 3684, `http://www.ietf.org/rfc/rfc3684.txt`, February 2004.

[49] Charles Perkins. IP Mobility Support for IPv4. IETF RFC 3344, `http://tool.ietf.org/rfc/rfc3344.txt`, August 2002.

[50] Charles Perkins, Elizabeth Belding-Royer, and Samir R. Das. Ad hoc On-Demand Distance Vector (AODV) Routing. IETF Experimental RFC 3561, `http://tool.ietf.org/rfc/rfc3561.txt`, July 2003.

[51] Charles Perkins, Elizabeth. Royer, and Samir R. Das. IP Address Autoconfiguration for Ad Hoc Networks. IETF Internet Draft, draft-ietf-manet-autoconf-01.txt, July 2000.

[52] Elizabeth Royer and Perkins Charles. Multicast Ad hoc Distance Vector (MAODV) Routing. IETF Internet draft, `http://tools.ietf.org/id/draft-ietf-manet-maodv-00.txt`, July 2000.

[53] Hesham Soliman, Claude Castelluccia, Karim El Malki, and Ludovic Bellier. Hierarchical Mobile IPv6 Mobility Management (HMIPv6). IETF Experimental RFC 4140, `http://www.ietf.org/rfc/rfc4140.txt`, August 2005.

[54] R Vida and Costa L. Multicast Listener Discovery Version 2 (MLDv2) for IPv6. IETF RFC 3810, `http://tool.ietf.org/rfc/rfc3810.txt`, June 2004.

[55] Ryuji Wakikawa, Jari Malinen, Charles Perkins, and Anders Nilsson. Global connectivity for IPv6 Mobile Ad Hoc Networks. IETF Internet draft, `http://tools.ietf.org/id/draft-wakikawa-manet-globalv6-01.txt`, July 2002.

[56] Ryuki Wakikawa, Jari Malinem, Charles Perkins, Anders Nilsson, and Antti J. Touminem. Global connectivity for IPv6 Mobile Ad Hoc Networks. IETF Internet Draft, draft-wakikwa-manet-globalv6-05.txt,, March 2006.

[57] Yunjung Yi, Sung-Ju Lee, William Su, and Mario Gerla. On-Demand Multicast Routing Protocol (ODMRP) for Ad Hoc Networks. IETF Internet draft, `http://tools.ietf.org/id/draft-ietf-manet-odmrp-04.txt`, November 2002.

Thesis

[58] João Paulo Barraca. Mecanismos de Facturação Segura em Redes Auto-Organizadas. Master's thesis, Universidade de Aveiro, December 2006.

Web Sites and Applications

[59] Ambient Networks. `http://www.ambient-networks.org`. FP6-IST-2002-507134.

[60] Designing Advanced Interfaces for the Delivery and Administration of Location independent optimized personal Services (DAIDALOS), EU IST Project. `www.ist-daidalos.org`. FP6-2002-IST-1-506997.

[61] WIP EU IST Project. `www.ist-wip.org`.

[62] Ad-hoc Network Autoconfiguration (autoconf). `http://www.ietf.org/html.charters/autoconf-charter.html`, June 2007.

[63] Ad hoc On-demand Distance Vector Routing. `http://moment.cs.ucsb.edu/AODV/aodv.html`, June 2007.

[64] APE testbed - APE Tools. `http://apetestbed.sourceforge.net/#APE_Tools`, June 2007.

[65] Host AP driver for Intersil Prims2. `http://hostap.epitest.fi/`, June 2007.

[66] Instituto de Telecomunicações - Aveiro. `http://www.av.it.pt`, June 2007.

[67] Mandriva Linux. `http://www.mandriva.com/`, June 2007.

[68] MGEN: The Multi-Generator Toolset.
`http://cs.itd.nrl.navy.mil/work/mgen/index.php`, June 2007.

[69] Mobile Ad-hoc Networks (manet) Charter.
`http://www.ietf.org/html.charters/manet-charter.html`, June 2007.

[70] MRD6, an IPv6 Multicast Router, June 2007.

[71] Netfilter/IPtables project. `http://www.netfilter.org`, June 2007.

[72] Network-based Localized Mobility Management (netlmm).
`http://www.ietf.org/html.charters/netlmm-charter.html`, June 2007.

[73] Uppsala University, Ad hoc implementation portal.
`http://core.it.uu.se/core/index.php/Main_Page`, June 2007.