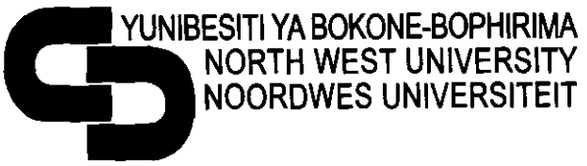

PERFORMANCE EVALUATION OF GPRS/802.11b MOBILE-NODE INITIATED HANDOVER BASED ON SIGNAL STRENGTH CRITERIA

SANDILE SONDAHA MTHOMBENI

B.ENG

11696761

Thesis submitted in partial fulfilment of the requirements for the degree Masters in Engineering at the North-West University

SUPERVISOR: PROF. A.S.J. HELBERG

NOVEMBER 2004

NORTH-WEST UNIVERSITY

Abstract

There has been a great concern by cellular providers about the impact Wireless Local Area Network (WLAN) systems that are based on 802.11b wireless standard pose to mobile business [24]. While cellular providers are still trying to offer high GPRS (General Packet Radio Service) data rates of more than 57kbps, 802.11b, also known as Wi-Fi, is already achieving more than 100 times 2.5G network data rates at 1 to 11Mbps. A user can enjoy high data rates using Wi-Fi in the comfort of a wireless connection to the Internet, but looses connection if he roams out of the range of the Wi-Fi Access Point (AP) covering the Mobile Node. In GPRS the user need not worry about the range shortcoming of the network, but the lower data rates of the connection to the Internet as well as the more expensive costs, which are shortcoming when compared to Wi-Fi. The apparent solution for fulltime coverage and high data rates at low cost during data communication is to integrate the two networks. This integration has proved to be a challenge for mobile operators in terms of offering mobility between 2.5G networks and 802.11b/a/g networks.

This thesis describes research followed to realize and evaluate performance of a handover mechanism based on signal strength criteria between GPRS and 802.11b access networks. The purpose is to integrate the two access networks with one Mobile Node (MN) and to evaluate the performance of a Mobile-Node initiated handover based on signal strength. This paper also describes the developed GPRS/802.11b testbed using Mobile IP standard (RFC 2002) to achieve handover between the hybrid networks. Mobile IP is an open standard still being refined by the IETF (Internet Engineering Task Force) to allow all IP based communication devices to roam from one network to the other.

Acknowledgments

This project would not be a reality without the support and contribution of the following people.

Prof. Albert Helberg

For his supervision, foresight, patience, motivation and guidance.

Niel Malan

For his support and friendship.

Ms. Mathuthu Mahlaba

My mother. Thank you for your teachings and setting an example.

University of North-West (Potchefstroom campus)

Academic resources.

Mrs. Klaasje Benadie

For assuring speedy delivery of project resources.

Telkom South Africa

For financial backup, and opportunity to study M.Eng.

Furthermore I would like to thank my sister Mapule and my nephew Mosehlane for their undivided love and support.

Contents

Figures

List of Tables

Abbreviations

BSC	Base Station Controller
BSS	Base Station Subsystem
BSSGP	BSS GPRS Protocol
BTS	Base Transceiver Station
CDMA	Code Division Multiple Access
CN	Correspondent Node
CSMA/CA	Carrier Sense Multiple Access with Collision Avoidance
CSMA/CD	Carrier Sense Multiple Access with Collision Detection
DCF	Data Communication Function
DHCP	Dynamic Host Configuration Protocol
DSSS	Direct Sequence Spread Spectrum
EIR	Equipment Identity Register
ETSI	European Telecommunications Standardization Institute
FA	Foreign Agent
FDMA	Frequency Division Multiple Access
FN	Foreign Network
GGSN	Gateway GPRS Support Node
GPRS	General Packet Radio Service
GRE	Generic Routing Encapsulation
GSM	Global System for Mobile communications
GSN	GPRS Support Node
GTP	GPRS Tunneling Protocol
GUI	Graphical User Interface
HA	Home Agent
HLR	Home Location Register
HN	Home Network
HO	Handover

ICMP	Internet Control Message Protocol
IEEE	Institute of Electrical and Electronics Engineers
IETF	Internet Engineering Task Force
IMEI	International Mobile Equipment Identity
IMSI	International Mobile Subscriber Identification
IP	Internet Protocol
IPv4	Internet Protocol version 4
IPv6	Internet Protocol version 6
ISO	International Organization for Standardization
LAN	Local Area Network
LLC	Logical Link Control
MAC	Medium Access Control
MH	Mobile Host
MIP	Mobile IP
MN	Mobile Node
MS	Mobile Station
MSC	Mobile Switching Centre
OFDM	Orthogonal Frequency Division Multiplexing
PAN	Public Area Network
PCI	Peripheral Component Interconnect
PCMCIA	Personal Computer Memory Card International Association
PDN	Packet Data Network
PDP	Packet Data Protocol
PLMN	Public Land Mobile Network
PPP	Point-to-Point Protocol
PSTN	Public Switched Telephone Network
QoS	Quality of Service
RF	Radio Frequency
RFC	Request for comment. An electronic document published on the Internet

RLC	Radio Link Control
SGSN	Serving GPRS Support Node
SNR	Signal-to-Noise Ratio
SPI	Security Parameter Index
TCP/IP	Transport Control Protocol/Internet Protocol
TDMA	Time Division Multiple Access
UDP	User Datagram Protocol
UMTS	Universal Mobile Telecommunications Systems
USB	Universal Serial Bus
WLAN	Wireless Local Area Network
WWAN	Wireless Wide Area Network
VLR	Visitors Location Register
VoIP	Voice-over-IP

1. Introduction

During the past years, telecommunication networks have emerged as a central strategic component in various fields. They are needed to form the worldwide infrastructure needed to support educational purposes, economic development, scientific research, and social interaction between people in various fields. The increase of high-speed local data networks and new multimedia services has driven the broadband telecommunication networks to be the focus of research, development, and standards activities worldwide. One of the accelerating factors is mobility. Mobility can be defined as a possibility for a user to use his network resources freely in any place and at anytime [3]. He can access remote databases and mailboxes by using lightweight Graphical User Interface (GUI) devices, e.g. laptops or palm pilots, anywhere and anytime.

1.1 Problem Statement

1.1.1 Introduction

Lately many Wireless Local Area Network (WLAN) systems have been installed in different kinds of areas such as hotels, coffee shops, airports, universities and offices. These areas are called Hot-Spots, which are mostly based on the Institute of Electrical and Electronics Engineers (IEEE) 802.11b standard to allow mobile users to achieve bandwidth of up to 11 Mbps. Hence, there is a great concern by cellular providers about the impact WLAN systems that are based on 802.11b wireless standard pose to mobile business [24]. While cellular providers are still trying to offer high General Packet Radio Service (GPRS) data rates of more than 57kbps, 802.11b, also known as Wi-Fi, is already achieving more than 100 times 2.5G network data rates at 1-11 Mbps. Wireless IP users of today, and eventually all consumers in the future, want to communicate and be able to do their daily business anytime and anywhere. As a result, there is a real demand for connectivity that is present everywhere between a wide variety of mobile devices and access technologies, which include GPRS and WLAN [1].

Incorporating mobility into broadband systems requires many considerations in every layer of the communication: power control in the physical layer, traffic management in the data link layer,

mobility management in the network layer and communication in the transport and application layer. Combining the services of telecommunication and data networks adds value to both by increasing usability and scalability. Both operators and software industry are looking for solutions that can be implemented with minimum changes to the already existing infrastructures [2].

1.1.2 Existing handover technology

There are different ways to implement handover in order to achieve mobility between GPRS and WLAN. Handover technologies that could be implemented consist of two basic areas, namely; network and consumer's equipment. The former solution results in significant changes in terms of addition and modification to the already existing cellular network. This solution is mostly proprietary and costly for mobile operators. The latter solution is to develop a Mobile Node (MN) e.g. laptop, cellular phone or PDA that allows the MN to initiate handovers between hybrid networks. This solution has manifested commercially in different ways. One of these ways is to implement a capability in the MN to access various access networks, but without the capability to do handover between them. Hence, the solution still results in disconnection if the consumer roams out of the range of the used network or even if he decides to use other access network supported by his MN.

With the introduction of Mobile IP, mobility and handovers are attainable in WLANs but only at the MAC layer. For instance, WLAN technology permits MAC handovers between networks by associating the MAC address of the mobile equipment with more than one Access Point that covers a WLAN infrastructure. The issue of achieving handover between hybrid networks at the network layer is still under investigation.

It was found from IEEE (Institute of Electrical and Electronics Engineers) articles and IETF (Internet Engineering Task Force) RFCs' literature study, that Mobile IP is the major technology that can be used to achieve handovers between hybrid networks. This technology enables layer 3 (IP) inter-system handovers between hybrid networks, thus implementing the mobility management in the network layer [3]. Because the mobility functions of Mobile IP are performed at the network layer rather than the physical layer, the mobile device can span different types of wireless and wireline networks while maintaining connections and ongoing applications as shown in *Figure 1.1*. At present, GPRS access is

available almost everywhere, thus Mobile IP can be used to fill the wireless access gap between disjunct 802.11b domains or even only between 802.11b and GPRS domains.

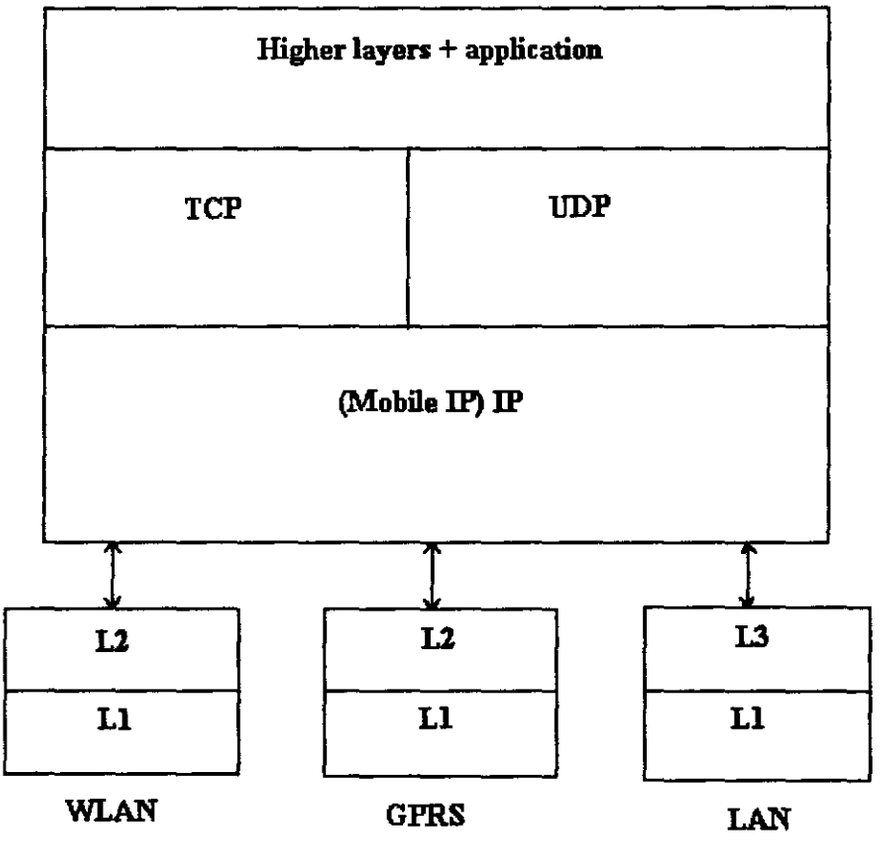

Figure 1.1 MIP Protocol stack for various interfaces

1.1.3 Handover issues

There are a variety of issues related to handovers, especially in data communication. These issues can be classified as: architectural and handover (HO) decision time algorithms. An important factor in the architectural issue is that of addressing which entity involved in the data communication controls the handover. The handover control can be located in the network entity either in GPRS or WLAN network or in the Mobile Node itself. These cases are called Network-Controlled Handover (NCHO) and Mobile-Controlled Handover (MCHO), respectively [3, 4].

The decision time algorithm issue manifests itself in unwanted or untimely handovers. A variety of metrics can be deployed to decide on the HO. The handover can be instigated due to different metrics like the Received Signal Strength (RSS), Signal to Noise Ratio (SNR), bandwidth degradation, and presence of a MN in another network in either of the networks. The handover decision metrics play an important role in handovers between GPRS and WLAN since they can affect data communication of the user by making unnecessary or uncomfortable handovers. This is termed the ping-pong effect [4], whereby a MN changes access network frequently in a short space of time that it is unacceptable to the end user. There is ongoing research on decision metrics and algorithms in order to achieve timely and comfortable handovers between hybrid data networks. The RSS is the primary decision metric that can be used in conjunction with various metrics to minimize the ping-pong effect in HO for hybrid data networks.

1.2 Purpose of the project

The research will focus on evaluating the performance of GPRS/802.11b Mobile-Node initiated handover based on signal strength criteria. A testbed network will be set-up to test the performance of RSS decision metric: the RSS metric will be used in combination with other metrics like dwell time and number of counts the current RSS remains below the minimum threshold. This is to improve the performance of the handover. The evaluation will be done on the performance measures of the handover rate and delay in HO. The decision metric or criterion that optimizes the performance measures and thus improves mobility will be concluded as the best method to achieve handover.

1.2.1 Sub-problems

There are various sub-problems that have to be addressed in order to achieve the purpose of the project. These sub-problems are discussed categorically in respect to the challenges that have to be faced in WLAN, GPRS and MIP. Linux operating system is selected as the platform to do the experiment since the MIP software used is developed for this operating system. This dictates the use of hardware and software drivers that are compliant to Linux.

WLAN

Received Signal Strength is used as a criterion to make a decision on when to initiate and end a handover. This is because other criteria closely relate to the changes in signal strength. These criteria are SNR and data link rate in the Wi-Fi. In this regard a WLAN card with drivers that can provide information on RSS, SNR and data link rate have to be used in the project. A WLAN card that is compliant to be used in Linux had to be found. Lucents'Orinoco silver WLAN card is used in this project since Linux provides a driver and updates for this card are available on the Internet. The use of the card in Linux is also well documented and its updated drivers can be downloaded from the Internet.

GPRS

A GPRS account has to be created and the GPRS card that can be set up in the Linux platform has to be found. The GPRS cards are developed to be used in the Windows operating system environment and their settings are for this operating system. The Option Wireless Technology's GPRS card is selected to be used as the GPRS interface. This is because literature indicated that although the card is solely developed for use in Windows environment there are unofficial configurations developed for the card to be used in GPRS, and help in setting the ppp configuration in Linux is available through email or telephone consultation with the configuration developers. This was the case in this project i.e. consultation in setting up the card was done with the help of the configuration developer via email.

Mobile IP

An open source Mobile IP stack had to found and Hut Dynamics 0.8.1 was selected. This choice eliminates high costs in attaining licences to use Mobile IP software. Hut Dynamics MIP software provides vast free available information on how to set it up. The software is free and it still being developed by the Hut Dynamics community on the Internet. The use and settings of the software was obtained from the user email group, which had and has similar problems in settings that were encountered in this project. It provided a solution to this project since the software requires daemons and configuration files to the Home Agent and Mobile Node, which are required in this project.

Performance Parameters

Measuring software that can be used in Linux had to be found and the RRDTool software was selected. This is because this software provides a graphical view of RSS, SNR, data link rate and traffic over the WLAN and GPRS interface. The software is free and can be downloaded from the Internet and is still being updated by the RRDTool developers. Other software like tcpdump and iptraffic were used to get a data view of the behaviour of the handover. The latter software packages are available as part of the Linux operating system.

1.3 Approach

The proposed research will entail the following:

- What is GPRS and 802.11b?
- Mobile IP principles
- Handover solutions
- Set up of GPRS and 802.11b handover demonstration
- Parameter measurements
- Conclusions

1.4 Research Methodology

Phase 1: GPRS and 802.11b standards overview

The first phase in the research will entail the following: What is GPRS and 802.11b? How do GPRS and 802.11b networks function? An overview of GPRS and WLAN architecture is given. What are protocols used in GPRS and 802.11b data sessions? Chapter 2 of this thesis is devoted to a literature study on both 802.11b and GPRS standard. The basic network and Mobile Station operation is also covered. A discussion on the data session establishment and termination is also discussed.

Phase 2: Mobile IP principles

Standard Mobile IP implementations and specifications as drafted by IETF will be studied. This will include a study of required Mobile IP network resources like the use of gateways and network topologies. This phase will also involve study of the ip routing, Agent Advertisement and Mobile Node's registration messages. Specific Mobile IP problems and tunneling modes are also discussed. How can the Mobile IP technology be implemented to achieve a handover between GPRS and 802.11b? Chapter 3 will be devoted to the implementation of Mobile IP.

Phase 3: Handover solutions

Knowledge acquired from the above two phases will be used to determine the following: What Mobile IP software package could be used? What network topologies are currently in use to achieve data session handover between the two networks? Handover controls will be analyzed in terms of their differences in handover initiation. Configuration of both Mobile Node and Home Agent's MIP software will be studied. Chapter 4 is dedicated to this phase of the research.

Phase 4: Handover demo set-up

A demonstration will be set up and configuration of MIP software scripts will be done with respect to handover initiation mechanism on both MN and HA. The handover initiation will be based on different signal strength criteria. Parameters to be studied are handover rate and handover delay. Comparison of RSS criteria will be done on the mentioned parameters. Chapter 5 is devoted to the demonstration set-up and comparison between the handover criteria.

Phase 6: Final conclusion

Phase 6 of the research is devoted to the results obtained in the above phase and a conclusion is given on the appropriate and feasible method of achieving data session handover between the networks. The overall success of the project as well as possible future research will be given and discussed. The results and conclusion are given in Chapter 6 and 7 respectively of the thesis.

1.5 Chapter summary

Chapter 1 focuses on the problem statement of the research as well as the basic approach that will be followed to realize completion of the project. The chapter briefly discussed both Mobile Node and network initiated handovers. A more detailed methodology that addresses issues and questions that need to be answered for the success of the project was outlined in this chapter. This included the use of Mobile IP to achieve handovers and achieve mobility between hybrid networks. Chapter 2 will give an introduction of GPRS and 802.11b standards as a background study for the research.

2. Background

2.1 Introduction

This chapter gives a background study on General Packet Radio Service (GPRS) and WLAN networks. It first gives an overview of the GPRS standard and discusses functionalities involved during data communication. The GPRS architecture is discussed in respect to additional entities to the Global System for Mobile communications (GSM) network. GPRS protocol is explained in terms of the layers that are involved during communication between GPRS network and packet data networks like Internet. The GPRS topic is then followed by an overview of the WLAN network. The WLAN topic first discusses the network with regard to its purpose, difference from normal LAN e.g. Ethernet and its architecture. It then explains the emergence of the IEEE 802.11 WLAN standard up to its evolution to 802.11b standard and beyond. The information presented in this chapter was collected from sources [5-9, 21].

2.2 What is GPRS?

GSM is a widely adopted cellular network technology in South Africa, European and more than 100 countries around the world. There existed great need and interest to add the capacity of data transmission to already existing GSM networks. However, the GSM network was designed to provide voice services, which caused the limitation in both maximum bit rate provided and the efficiency when handling data instead of voice. It was therefore necessary to introduce packet switching in the existing GSM networks in order to provide an attractive bearer service for users desiring fast, efficient and affordable access to the Internet and their corporate intranet. Extending GSM networks to support packet switched services was therefore critical for mobile operators wanting to position themselves in the 3rd Generation market. GSM using circuit switch data technology as a data bearer is inferior for it has been optimized for speech. For data, it provides a mere 9.6 kbps on the air-interface [7].

In order to address these inefficiencies of cost and data rate, cellular packet data technologies were developed with one of them being GPRS. GPRS was originally developed for GSM, but it is also

being integrated in other cellular standards. GPRS is a new bearer service for GSM that improves and simplifies wireless access to packet data networks, e.g. Internet. It applies a packet radio principle to transfer user data packets in an efficient way between GSM Mobile Stations and external packet data networks. Packets can be directly routed from the GPRS Mobile Stations to packet switched networks. Networks based on the Internet Protocol (IP) networks are supported in the current version of GPRS.

Users of GPRS benefit from shorter access times and higher data rates as compared to conventional GSM where the connection setup takes seconds and rates for data transmission are restricted to 9.6 kbps. GPRS in practice offers session establishment times of less than a second and theoretical data rates of up to 115kbps, with a practical 57 kbps data rate. In addition, GPRS packet transmission offers a user friendlier billing than that offered by circuit switched services that was originally used by GSM [7, 8]. In circuit switched services, billing is based on the duration of the connection. This is unsuitable for applications with bursty traffic. The user must pay for the entire airtime, even for idle periods when no packets are sent (e.g. when the user reads a Web page). In contrast to this, with packet switched services, billing can be based on the volume of transmitted data. The advantage for the user is that he or she can be online over a long period but will only be billed on the transmitted data volume.

To summarize GPRS improves the utilization of the radio resources, offers volume-based billing, higher transfer rates, shorter access times, and simplifies the access to packet data networks. GPRS has been standardized by European Telecommunications Standards Institute (ETSI) during the last six years. It finds great interest among many GSM network providers. GPRS is being implemented in various countries including South Africa whereby operators are already in the process of evolving to 3G networks.

2.3 GPRS architecture [8, 10]

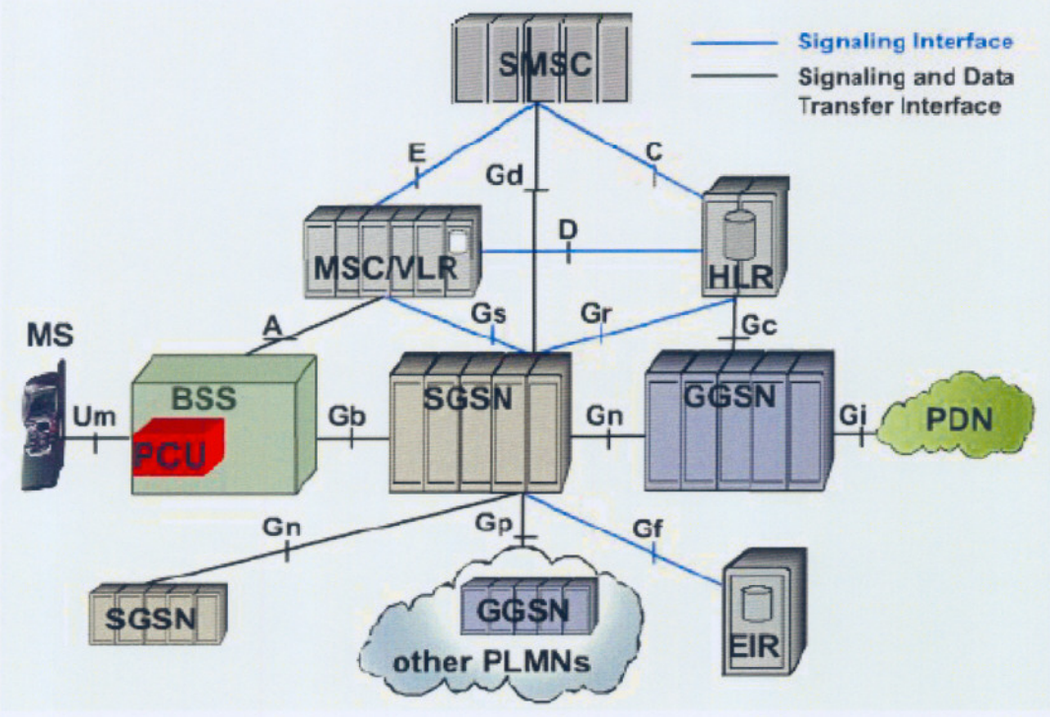

Figure 2.1 GPRS Architecture [8]

A new class of network nodes called GPRS support nodes (GSN) were integrated into the existing GSM architecture. These nodes are the serving GPRS support node (SGSN) and the gateway GPRS support node (GGSN). GSNs are responsible for delivering and routing of data packets between MNs and the external packet data networks (PDN). *Figure 2.1* illustrates the system architecture. These nodes interwork with the home location register (HLR), the mobile switching centre/visitor location register (MSC/VLR) and the base station subsystem (BSS). The following subsections will describe basic function of these mentioned nodes [8].

2.3.1 Serving GPRS Support Node (SGSN)

The SGSN physical entity is in general responsible for the communication between the GPRS network and all the GPRS users located within its service area (service area is discussed in section 2.3.1.1). Some of the functions performed by the SGSN are:

- to perform mobility management for GPRS terminals (attach/detach, user authentication, ciphering and location management
- to support combined mobility management for Class A and Class B mobile terminals by interworking with MSC/VLR
- to manage the logical link to mobile terminals (the logical link carries user packet traffic, SMS traffic, and layer signaling between the network and GPRS terminals)
- to route and transfer packets between the mobile terminals and GGSN
- to handle packet data protocol (PDP) context (the PDP context defines important parameters, such as the Access Point name, quality of service and the GGSN to be used for the connection to the external packet data networks)
- to interwork with the radio resource management in the BSS
- to generate charging data

2.3.2 Gateway GPRS support node (GGSN)

The GGSN is connected via an IP backbone to the SGSN and serves as interconnection point for packet data networks to the GPRS backbone network. The access-server functionality in the GGSN is defined according to standards from the Internet Engineering Task Force (IETF). The main functions of GGSN are:

- to convert the GPRS packets coming from the SGSN into the appropriate packet data protocol (PDP) format (e.g. IP), and sends them out on the corresponding PDN. In the other direction, PDP addresses of incoming data packets are converted to the GSM address of the destination user (the addressed packets are sent to the responsible SGSN, for this purpose, the GGSN stores the current SGSN address of the user and his or her profile in its location register.

- to authenticate users to external packet networks

- to generate charging data

2.3.3 Home Location Register (HLR)

The HLR stores the user profile, the current SGSN address, and the PDP address for each user in the public land mobile network (PLMN). The Gr is used to exchange this information between HLR and the SGSN. For example, the SGSN informs the HLR about the current location of the MN. When the MN registers with a new SGSN, the HLR will send the user profile to the new SGSN. The signalling path between GGSN and HLR (Gc interface) may be used by the GGSN to query a user's location profile in order to update its location register.

2.3.4 Visitors Location Register (VLR)

A VLR is responsible for a group of location areas and stores data of those users who are currently in its area of responsibility. This includes parts of the permanent user data that have been transmitted from the HLR to the VLR for faster access. However, the VLR may also assign and store local data such as a temporary identification of the MN.

2.3.5 Equipment Identity Register (EIR)

The equipment identity register (EIR) is a database that stores the unique International Mobile Station Equipment Identity (IMEI) of the MN. The IMEI is allocated by the manufacture, registered by the network operator, and stored in the EIR. Each registered user is identified by the international mobile subscriber identity (IMSI). It is stored in the subscriber identity module (SIM). The MN can only be operated if a SIM with a valid IMSI is inserted into equipment with a valid IMEI. It prevents calls from unauthorized or stolen MNs.

2.4 GPRS Protocol

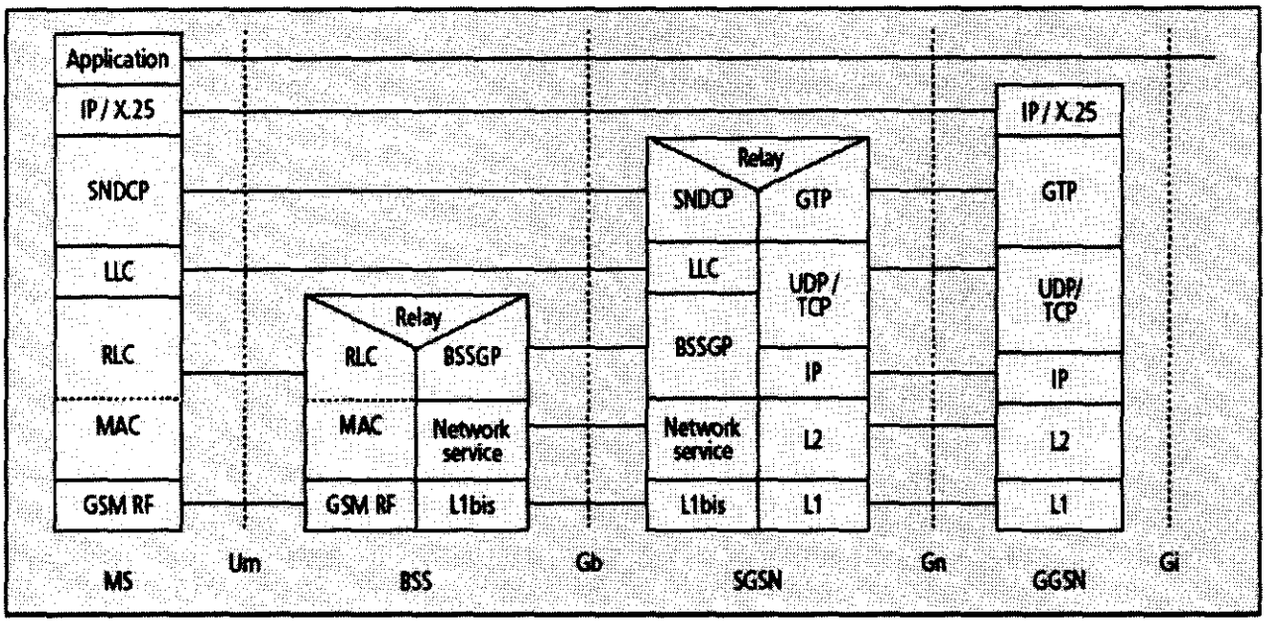

Figure 2.2 GPRS transmission plane [10]

The transmission plane is used to transfer the user data information among the different GPRS physical nodes. The protocol stack used by the GPRS transmission plane is shown in *Figure 2.2.* The following protocol layers can be identified and their functions are explained [9, 10]:

- The Application layer transfers application based information among end points (e.g., MS).
- IP (or X.25) layers are used as network layers.
- Tunnel Protocol (GTP) enables tunneling multiprotocol data packets through the GPRS backbone between GPRS support nodes (i.e., GGSN, SGSN).
- SNDCP provides data compression (e.g., V.42 bis) and header compression (e.g., TCP/IP header compression) in order to improve channel efficiency. The logical link control (LLC) protocol operates across the Gb and the Um interface, providing a logical link between the

MS and its SGSN [7]. Typical LLC functions comprise ciphering, flow control, and sequence control. In addition, if the LLC protocol is used in acknowledged mode, it provides detection and recovery of transmission errors; in unacknowledged mode it signals unrecoverable errors.

- The Base Station System GPRS Protocol (BSSGP) layer performs the transfer of QoS related information and routing between BSS and the SGSN physical nodes. This layer does not perform error correction.

- The Radio Link Control (RLC) / Medium Access Control (MAC) layer consists of the RLC and MAC sub-layers. RLC provides a reliable radio link and the MAC function is to control radio access signaling procedures, e.g., request and grant. Furthermore, it maps the LLC frames onto the GSM channels.

- In the BSS and SGSN physical nodes, interworking functions (Relays) between the RLC and BSSGP and between SNDCP and GTP respectively are required.

- The BSSGP packet data units (PDU) are transferred between the BSS and SGSN by the Network Service (NS) layer, and is based on Frame Relay technology.

- The GPRS radio physical consists of two layers. One of them is the Physical Link Layer (PLL) that is providing radio physical channels between the MS and BSS. The other layer is the Physical RF layer (RFL) and is mainly providing modulation and demodulation.

2.5 GPRS Channels [5, 6, 7]

On the physical layer, GSM uses a combination of Frequency Division Multiple Access (FDMA) and Time Division Multiple Access (TDMA) for multiple accesses. Two frequency bands are reserved for GSM operation, one for transmission from the Mobile Station to the BTS (uplink) and one for transmission from the BTS to the Mobile Station (downlink). Each of these bands is divided into 124 single carrier channels of 200 kHz width. A certain number of these frequency channels are allocated to a BTS, i.e., to a cell. Each of the 200 kHz frequency channels is divided into eight time slots that form a TDMA frame as shown in *Figure 2.3*. A time slot lasts for duration of 0.577 ms and carries 114 bits of information. The recurrence of one particular time slot defines a physical channel. Physical channels allocated for GPRS are called Packet Data Channels (PDCH) [5].

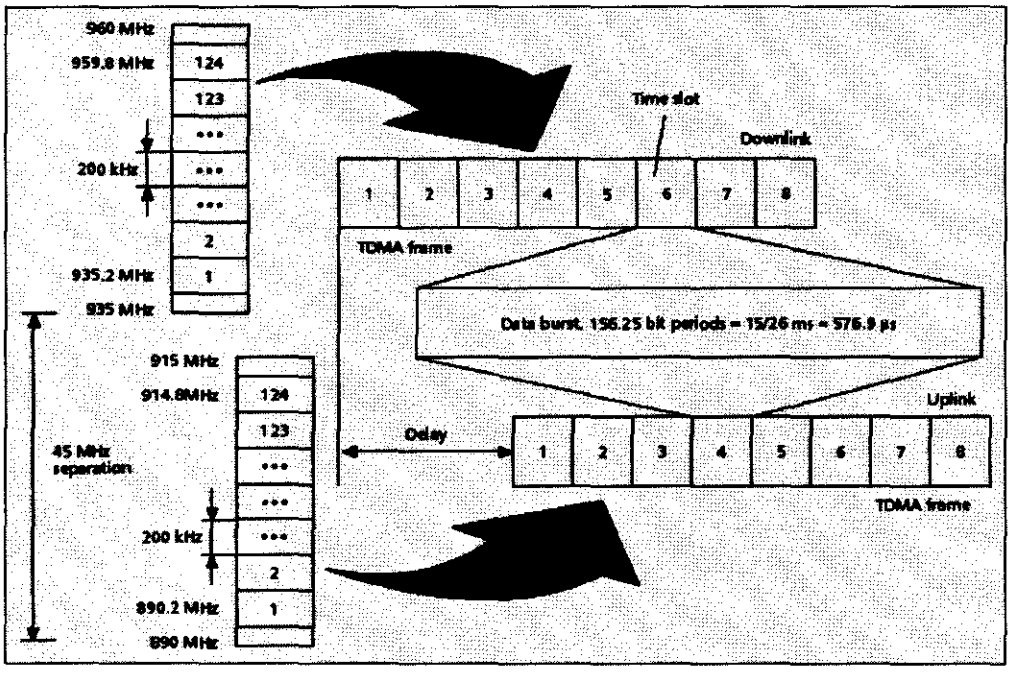

Figure 2.3 GPRS Channels [7]

The channel allocation in GPRS is different from the original allocation scheme of GSM. GPRS allows a single Mobile Station to transmit on multiple time slots of the same TDMA frame. This result in a very flexible channel allocation: one to eight time slots per TDMA frame can be

allocated to one Mobile Station. On the other hand a time slot can be assigned temporarily to a Mobile Station, so that one to eight Mobile Stations can use one time slot. Moreover, uplink and downlink channels are allocated separately, which efficiently supports asymmetric data traffic.

In conventional GSM, a channel is permanently allocated for a particular user during the entire call duration even when data is not transmitted. This is not the case in GPRS, the channels are only allocated when data packets are sent or received, and they are released after the transmission. For bursty traffic this results in a much more efficient usage of the scarce radio resource. With this principle, multiple users can share one physical channel. GPRS includes the functionality to increase or decrease the amount of radio resources allocated to GPRS on a dynamic basis. The PDCHs are taken from the common pool of all channels available in the cell. Physical channels that are not currently in use by conventional GSM can be allocated as PDCHs to increase the quality of service for GPRS. When there is a resource demand for services with higher priority, e.g. GSM voice calls, PDCHs can be de-allocated [6].

2.6 GPRS attach and detach procedure

A Mobile Node must first register with a SGSN of the GPRS network before it can use the services provided by GGSN. The network will check if the user is authorized, copy the user profile from the HLR to the SGSN and assign a packet temporary mobile subscriber identity (P-TMSI). This procedure is called the GPRS attach. The disconnection from the GPRS network is called GPRS detach. It can be initiated by the Mobile Node or by the network (SGSN or HLR) [6].

2.7 PDP Context [7]

To exchange data packets with external PDNs after a successful GPRS attach, a MN must apply for one or more addresses used in the PDN, e.g., for an IP address in case the PDN is an IP network. This address is called PDP address (Packet Data Protocol address). For each session a so called PDP context is created as shown in *Figure 2.4*, which describes the characteristics of the session. It contains the PDP type (e.g. IPv4), the address assigned to the MN e.g. IP address, the quality of service QoS, and the address of the GGSN that serves as the Access Point to the PDN. This context is stored in the MN, SGSN, and the GGSN.

With an active PDP context, the Mobile Node is visible for the external PDN and is able to send and receive data packets. The mapping between the two addresses, PDP and IMSI, enables the GGSN to transfer data packets between PDN and MN. A user may have several simultaneous PDP contexts active at a given time. The allocation of the PDP address can be static or dynamic. In the first case, the network operator of the user's home-PLMN permanently assigns a PDP address to the user. In the second case, the PDP address is assigned to the user upon the activation of a PDP context. The PDP address can be assigned to the user's home-PLMN (dynamic home-PLMN PDP address) or by the operator of the visited network (dynamic visited-PLMN PDP address). The Home Network operator decides which of the possible alternatives may be used. In case of dynamic PDP address assignment, the GGSN is responsible for the allocation and the activation/deactivation of the PDP addresses.

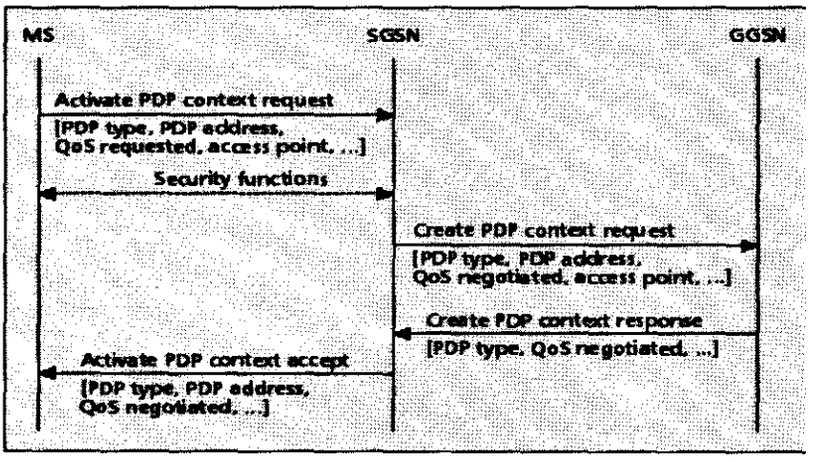

Figure 2.4 PDP Context activation [7]

Figure 2.4 shows the PDP context activation procedure. Using the message "activate PDP context request," the MS informs the SGSN about the requested PDP context. If dynamic PDP address assignment is requested, the parameter PDP address will be left empty. Afterward, usual security functions e.g. authentication of the user are performed. If access is granted, the SGSN will send a "create PDP context request" message to the affected GGSN. The latter creates a new entry in its PDP. Afterward, the GGSN returns a confirmation message "create PDP context response" to the SGSN, which contains the PDP address in case dynamic PDP address allocation was requested. The

SGSN updates its PDP context tables and confirms the activation of the new PDP context to the MN ("activate PDP context accept").

2.8 GPRS routing mechanism [5]

A scenario of how packets are routed in GPRS can be explained as follows: it is assumed that the packet data network is an IP network. A GPRS Mobile Station located in PLMN sends IP packets to a host connected to the IP network, e.g., to a Web server connected to the Internet. The SGSN that the Mobile Station is registered with encapsulates the IP packets coming from the Mobile Station, examines the PDP context, and routes them through the intra-PLMN GPRS backbone to the appropriate GGSN. The GGSN decapsulates the packets and sends them out on the IP network, where IP routing mechanisms are used to transfer the packets to the access router of the destination network. The latter delivers the IP packets to the host.

2.9 Location Management

The main task of location management is to keep track of the user's current location, so that incoming packets can be routed to his or her MS. For this purpose, the MS frequently sends location update messages to its current SGSN. If the MS sends updates rather seldom, its location (e.g., its current Cell) is not exactly known and paging is necessary for each downlink packet, resulting in a significant delivery delay. On the other hand, if location updates happen very often, the MS's location is well known to the network, and the data packets can be delivered without any additional paging delay. However, quite a lot of uplink radio capacity and battery power is consumed for mobility management in this case. Thus, a good location management strategy must be a compromise between these two extreme methods [8].

For this reason, a state model shown in *Figure 2.5* has been defined for location management in GPRS. An MS can be in one of three states depending on its current traffic amount; the location update frequency is dependent on the state of the MS.

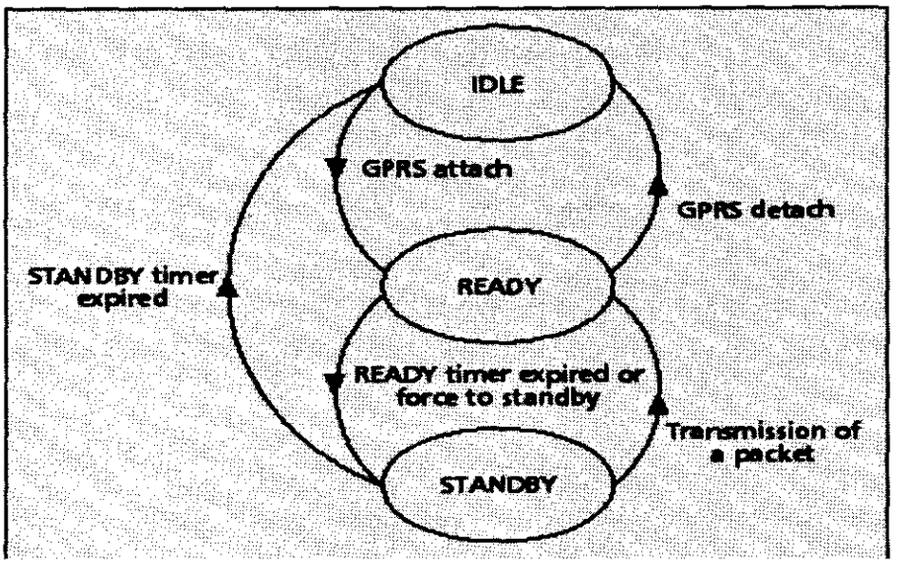

Figure 2.5 State Model of a GPRS Mobile Station [7]

In IDLE State the MS is not reachable. Performing a GPRS attach, the MS gets into READY state. With a GPRS detach it may disconnect from the network and fall back to IDLE state. All PDP contexts will be deleted. The STANDBY state will be reached when an MS does not send any packets for a longer period, when the READY timer (which was started at GPRS attach) expires.

In IDLE State, no location updating is performed, i.e., the current location of the MS is unknown to the network. An MS in READY State informs its SGSN of every movement to a new cell. For the location management of an MS in STANDBY State, a GSM location area (LA) is divided into several routing areas (RA). In general, a RA consists of several cells. The SGSN will only be informed when an MS moves to a new RA; cell changes will not be disclosed. To find out the current cell of an MS in STANDBY State, paging of the MS within a certain RA must be performed. For MSs in READY State, no paging is necessary [7, 8].

Whenever an MS moves to a new RA, it sends a "routing area update request" to its assigned SGSN. The message contains the routing area identity (RAI) of its old RA. The base station subsystem (BSS) adds the cell identifier (CI) of the new cell, from which the SGSN can derive the new RAI. Two different scenarios are possible:

Intra-SGSN routing area update: The MS has moved to an RA that is assigned to the same SGSN as the old RA. In this case, the SGSN has already stored the necessary user profile and can assign a new packet temporary mobile subscriber identity (P-TMSI) to the user ("routing area update accept"). Since the routing context does not change, there is no need to inform other network elements, such as GGSN or HLR.

Inter-SGSN routing area update: The new RA is administered by a different SGSN than the old RA. The new SGSN realizes that the MS has changed to its area and requests the old SGSN to send the PDP contexts of the user. Afterward, the new SGSN informs the involved GGSNs about the user's new routing context. In addition, the HLR and (if needed) the MSC/VLR are informed about the user's new SGSN.

2.10 Interworking with IP networks

This section will show how a GPRS network can be interconnected with an IP-based packet data network, such as the Internet or intranets. GPRS supports both IPv4 and IPv6. As shown in *Figure 2.7* the Gi interface is the interworking point with IP networks. From outside, i.e., from an external IP network's point of view, the GPRS network looks like any other IP subnetwork, and the GGSN looks like a usual IP router. *Figure 2.6* gives an example of how a GPRS network may be connected to the Internet. Each registered user who wants to exchange data packets with the IP network gets an IP address, as explained earlier. The IP address is taken from the address space of the GPRS operator [7].

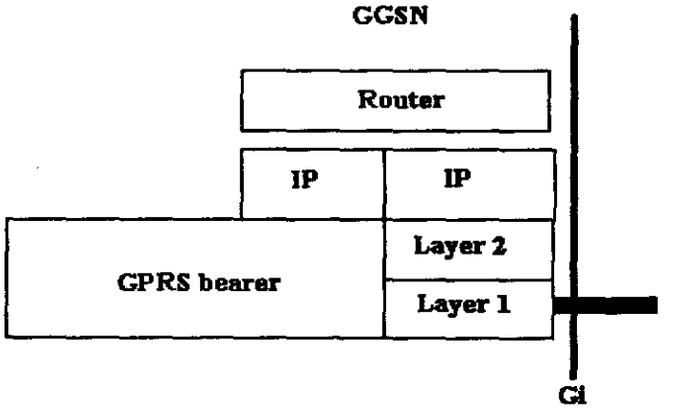

Figure 2.6 Protocol at the Gi IP interface

In order to support a large number of mobile users, it is essential to use dynamic IP address allocation (in IPv4). Thus, a DHCP server (Dynamic Host Configuration Protocol) is installed. The address resolution between IP address and GSM address is performed by the GGSN, using the appropriate PDP context. The routing of IP packets and the tunneling through the intra-PLMN backbone (using the GPRS Tunneling Protocol GTP) has been explained in prior sections. Moreover, a domain name server (DNS) managed by the GPRS operator or the external IP network operator can be used to map between external IP addresses and host names. To protect the PLMN from unauthorized access, a firewall is installed between the private GPRS network and the external IP network. With this configuration, GPRS can be seen as a wireless extension of the Internet all the way to a Mobile Station or mobile computer. The mobile user has direct connection to the Internet [7, 21].

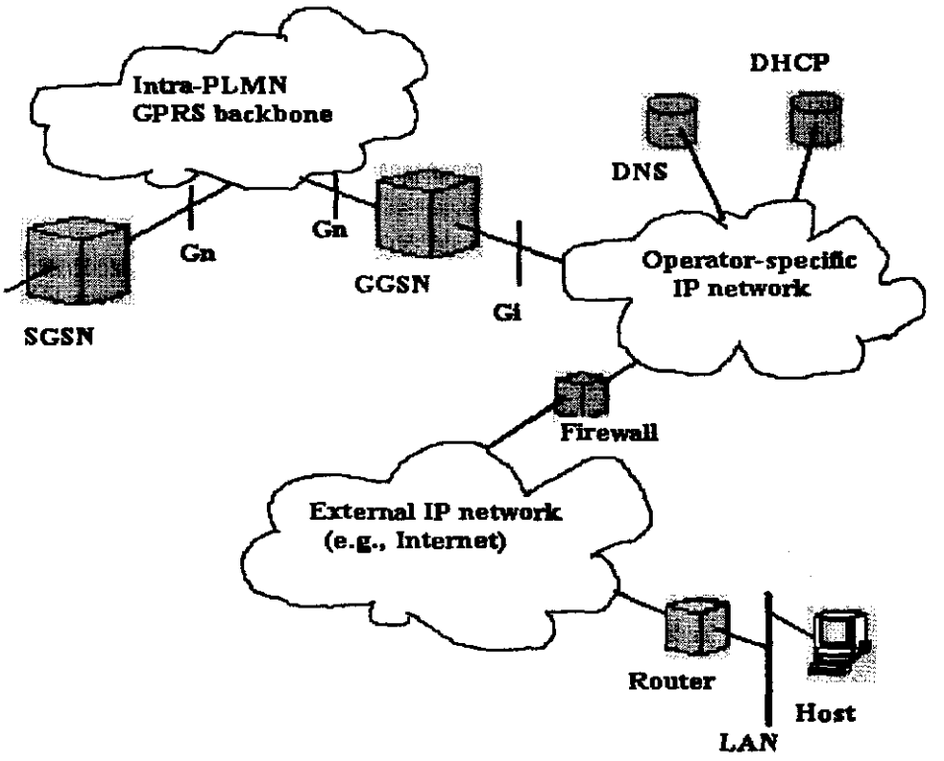

Figure 2.7 GPRS Internet connection

2.11 What is a WLAN?

A Wireless Local Area Network (WLAN) is a flexible data communications system that can either replace or extend a wired LAN to provide added functionality. Using Radio Frequency (RF) technology, WLANs transmit and receive data over the air, through walls, ceilings and even cement structures, without wired cabling. A WLAN provides all the features and benefits of traditional LAN technologies like Ethernet and Token Ring, but without the limitations of being wired to a cable. This provides greatly increased freedom and flexibility. Just as wired LANs use copper or fiber optic cable, WLANs provide data connectivity between computing devices by using radio waves as the carrier medium instead of a physical cable infrastructure. Data is superimposed onto a radio wave through a process called modulation, and this "carrier wave" then acts as the transmission medium, taking the place of a wire.

2.12 IEEE 802.11 protocol stack [26]

The most prominent specification for wireless LANs was developed by IEEE 802.11 working group, with charter to develop a MAC (Medium Access Control) protocol and physical medium specification as shown in *Figure 2.8*. For compatibility purposes, the 802.11 MAC must appear to the upper layers of the network as a standard 802 LAN. The 802.11 MAC layer is forced to handle station mobility in a fashion that is transparent to the upper layers of the 802 LAN stack. This forces functionality into the 802.11 MAC layer that is typically handled by upper layers [9].

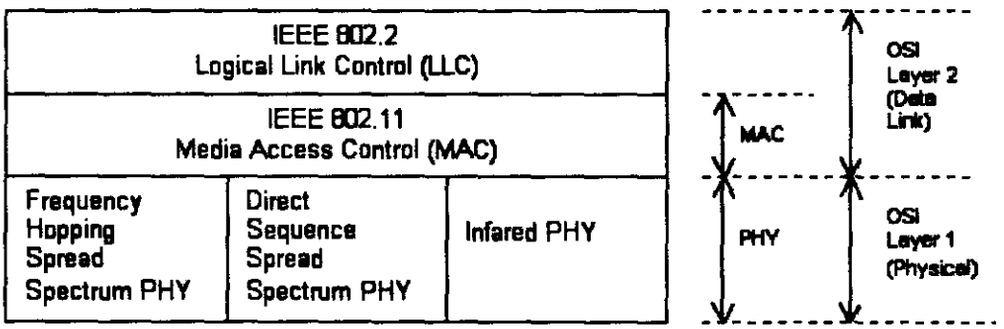

Figure 2.8 802.11 Protocol stack

2.12.1 IEEE 802.11 Media Access Control [9, 12]

The IEEE 802.11 MAC layer covers three functional areas which are reliable data delivery, access control and security. These functions are discussed respectively below as follows:

- The 802.11 MAC layer provides functionality to allow reliable data delivery for the upper layers over the wireless PHY media. The data delivery itself is based on an asynchronous, best-effort, connectionless delivery of MAC layer data. There is no guarantee that the frames will be delivered successfully.

- The 802.11 MAC provides a controlled access method to the shared wireless media called Carrier-Sense Multiple Access with Collision Avoidance (CSMA/CA). CSMA/CA is similar to the collision detection access method deployed by 802.3 Ethernet LANs.

- The third function of the 802.11 MAC is to protect the data being delivered by providing security and privacy services. Security is provided by the authentication services and by Wireless Equivalent Privacy (WEP), which is an encryption service for data delivered on the WLAN.

2.12.2 IEEE 802.11 Physical Layer (PHY)

The 802.11 physical layer (PHY) is the interface between the MAC and the wireless media where frames are transmitted and received. The PHY provides three functions. First, the PHY provides an interface to exchange frames with the upper MAC layer for transmission and reception of data. Secondly, the PHY uses signal carrier and spread spectrum modulation to transmit data frames over the media. Thirdly, the PHY provides a carrier sense indication back to the MAC to verify activity on the media.

802.11 standard provides three different PHY definitions: Both Frequency Hopping Spread Spectrum (FHSS) and Direct Sequence Spread Spectrum (DSSS) support 1 and 2 Mbps data rates. An extension to the 802.11 architecture (802.11a) defines different multiplexing techniques that can

achieve data rates up to 54 Mbps. Another extension to the standard (802.11b) defines 11 Mbps and 5.5 Mbps data rates (in addition to the 1 and 2Mbps rates) utilizing an extension to DSSS called High Rate DSSS (HR/DSSS). 802.11b also defines a rate shifting technique where 11 Mbps networks may fall back to 5.5 Mbps, 2 Mbps, or 1 Mbps under noisy conditions or to inter-operate with legacy 802.11 PHY layers [9].

2.12.2.1 IEEE 802.11 Radio Transmission Technology [10]

A brief explanation of these signal-spreading techniques is as follows:

Frequency Hopping Spread Spectrum (FHSS) was originally conceived as a means to hide a transmission from unwanted listeners. It is now utilized for another purpose, the reduction of interference. Frequency hopping works by transmitting the signal carrier for a short period of time on one narrow band, then hopping to another. Over a period of time, the average signal power is thus spread over a very wide band of frequencies.

The frequency hops appear random to anyone who doesn't know the pre-arranged hop pattern. Fifty years ago this made it impossible to tune in and listen to a transmission, because the signal carrier never stayed on one frequency long enough for the listener to locate it and retune the receiver to the new frequency. Today, wireless LANs that incorporate FHSS do so on predetermined hopping sequences that are not secret, and the technology to follow the hopping pattern and retrieve the signal is available for the cost of a wireless card. Thus, FHSS, as employed by wireless LANs and PANs, no longer offers any inherent security. It does, however, serve to reduce interference to and from other devices.

Direct Sequence Spread Spectrum (DSSS) is a more complex technique which spreads the signal's power across a wider bandwidth by spreading the carrier itself, instead of rapidly moving it around as FHSS does. It does this by directly modulating the carrier with a high-speed code sequence, which has the characteristics of pseudo-random noise (PN). The faster a carrier is modulated, the wider its bandwidth becomes.

The spreading sequence is produced by modulating the data stream with a PN spreading code, thus resulting in a signal which has a much higher bandwidth than the information bandwidth

alone. For example, with 1 and 2 Mbps 802.11 DSSS, each bit of data is logically combined with an 11-bit Barker code. Because the bit-rate (chip-rate) of the spreading sequence is much higher than that of the data rate, the bandwidth is effectively spread over a much larger area than would otherwise be occupied if the carrier was modulated by the data stream alone. The result is that the signal power is spread over a much larger band and appears to other users as low-power noise.

Orthogonal Frequency Division Multiplexing (OFDM), which utilizes multiple carriers (referred to as subcarriers), is technically not a spread spectrum technique because the subcarriers remain stationary and are not spread, but it serves the same purpose of spreading the signal power over a large band. It does this by breaking the signal up into parts and transmitting each of the parts on a different subcarrier at a different center frequency. Thus a fast transmission is sent as many slow transmissions, simultaneously, on many different frequencies.

This effective slowing of the symbol transmission rate, without slowing the actual data transmission rate, makes OFDM resistant to intersymbol interference resulting from multipath. In theory, if higher data rates are required, then the signal can just be broken up into more parts and transmitted on additional subcarriers, each part still being sent at a slow enough rate to avoid intersymbol interference.

These different techniques for spreading the signal's carrier, and the different digital modulation techniques employed to put information on the carrier, are central to defining the different wireless technologies and standards, as well as putting a perspective on interference issues among 802.11 LANs and Bluetooth networks.

2.13 WLAN equipment

Two main components form the basis of the wireless network. These components are the Mobile Node with a LAN adapter and an Access Point. The Orinoco Silver WLAN adapter and MSI 802.11b Access Point were used for this project.

2.13.1 MN/WLAN adapters

The Mobile Node (MN) or Mobile Host (MH) is the most basic component of the wireless network. A mobile host is any device that contains the functionality of the 802.11 protocol, that being MAC, PHY, and a connection to the wireless media. Typically the 802.11 functions are implemented in the hardware and software of a wireless network interface card (NIC) or wireless LAN adapter. A Mobile Node could be a laptop PC or handheld device. Stations may be mobile, portable, or stationary and all stations support the 802.11 station services of authentication, de-authentication, privacy, and data delivery. Wireless adapters are made in the same basic form factors as their wired counterparts PCMCIA, Cardbus, PCI and USB. They also serve the same function, enabling end users to access the network. In a wired LAN, adapters provide the interface between the network operating system and the wire. In a WLAN, they provide the interface between the network operating system and an antenna, to create a transparent connection to the network.

2.13.2 Access Point

Essentially, the Access Point (AP) is the wireless equivalent of a LAN hub. It receives, buffers and transmits data between the WLAN and the distribution system, supporting a group of wireless user devices. An Access Point is typically connected with the wired backbone through a standard Ethernet cable, and communicates with wireless devices by means of an antenna. The Access Point, or the antenna connected to it, is generally mounted high on a wall or on the ceiling. Like the cells in a cellular phone network, multiple Access Points can support handover from one Access Point to another as the user moves from area to area.

Access Points have ranges from under 20 meters to 500 meters, and a single Access Point can support between 15 and 250 users, depending on the technology, configuration and use. It is relatively easy to scale WLANs by adding more Access Points. This decreases network congestion and enlarges the coverage area. Large facilities requiring multiple Access Points deploy them to create overlapping cells for constant connectivity to the network. A wireless Access Point can track movement of clients across its domain and permit or deny specific traffic or clients from communicating through it.

2.14 WLAN architecture [11]

An 802.11 WLAN is based on a cellular architecture where the system is subdivided into cells. Each cell called Basic Service Set, or BSS, in the 802.11 nomenclature is controlled by a Base Station called Access Point (AP). Although a wireless LAN may be formed by a single cell, with a single Access Point, (and as will be described later, it can also work without an Access Point), most installations will be formed by several cells, where the Access Points are connected through some kind of backbone called Distribution System (DS). This backbone is typically Ethernet and, in some cases, is wireless itself. The whole interconnected Wireless LAN, including the different cells, their respective Access Points and the Distribution System, is seen as a single 802 network to the upper layers of the OSI model and is known in the Standard as Extended Service Set (ESS) [11]. A WLAN can be configured in two basic ways: Independent Basic Service Set (IBSS) Peer-to-Peer and Infrastructure Basic Service Set also called the Client/Server infrastructure.

2.14.1 Independent Basic Service Set (IBSS)

The most basic wireless LAN topology is a set of stations, which have recognized each other and are connected via the wireless media in a peer-to-peer fashion. This form of network topology is referred to as an Independent Basic Service Set (IBSS) or an Ad-hoc network. This topology consists of two or more PCs equipped with wireless adapter cards, but with no connection to a wired network as shown in *Figure 2.9.* It is principally used to quickly and easily set up a WLAN where no infrastructure is available, such as at a convention center or offsite meeting location.

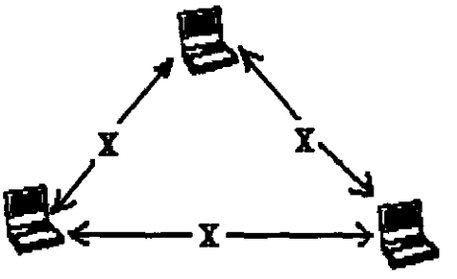

Figure 2.9 Peer-to-Peer WLAN architecture

2.14.2 Infrastructure Basic Service Set (BSS)

Infrastructure Basic Service Set or Client/server architecture offers distributed data connectivity, this mode typically consists of multiple PCs linked to a central hub that acts as a bridge to the resources of the wired network as depicted in *Figure 2.10.* In infrastructure WLANs, multiple Access Points link the WLAN to the wired network and allow users to efficiently share network resources. This is called the infrastructure basic service set. The Access Points not only provide communication with the wired network but also mediate wireless network traffic in the immediate neighborhood. Multiple Access Points can provide wireless coverage for an entire building or campus [10].

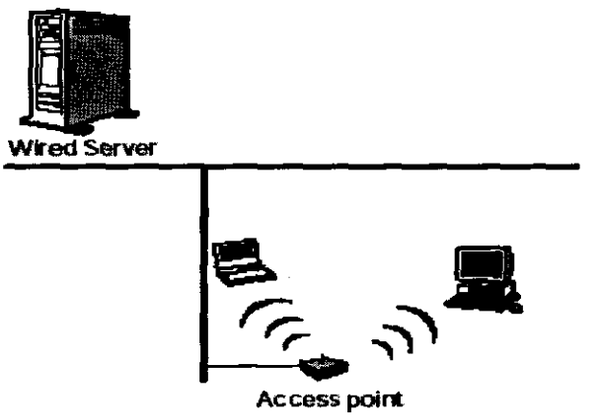

Figure 2.10 Client/Server WLAN architecture

An Extended Service Set (ESS) shown in *Figure 2.11* is formed when multiple overlapping BSSs (each containing an AP) are connected together by means of a distribution system, usually a wired Ethernet LAN. BSSs whose ranges overlap must transmit on different channels to avoid interference.

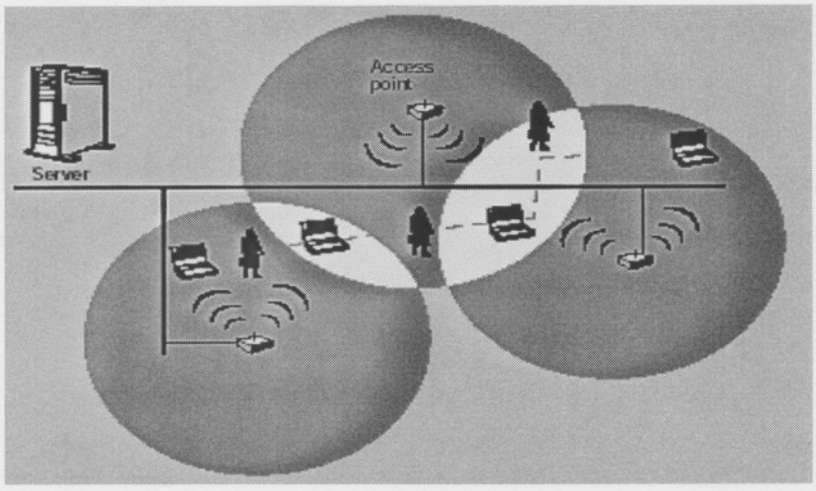

Figure 2.11 Extended Service Set architecture

The range between MNs and APs is up to 100 m (depending on data rate), but the overall range of an ESS is limited only by the range of the wired distribution system. Also, ESSs can be further extended with wireless links up to several miles by the use of directional range extender antennas.

2.15 IEEE 802.11 standards

Just as IEEE 802.3 Ethernet has evolved to become the predominant wired LAN technology, the IEEE 802.11 standard has also emerged as predominant WLAN. Like all IEEE 802 standards, the 802.11 standard focuses on the bottom two levels of the ISO model, the physical layer and data link layer. Any LAN application, network operating system or protocol, including TCP/IP, can run on 802.11 compliant WLAN as easily as they run over Ethernet. The 802.11 standard allow for three types of transmissions, which are: Frequency Hopping Spread Spectrum (FHSS), Direct Sequence Spread Spectrum (DSSS) and Orthogonal Frequency Division Multiplexing (OFDM) [10, 25].

2.15.1 Emergence of 802.11b products

The question that is startling in regard to 802.11 standards is why 802.11b products came before 802.11a. The letters after the number "802.11" indicate the order in which standards were first

proposed, not the order in which products appear. The first wireless LAN standard, 802.11, was approved by the Institute of Electrical and Electronics Engineers (IEEE) in 1997 and supported speeds up to 2 Mbps. In 1999, the IEEE approved both the 802.11a and 802.11b standards. 802.11a specified radios transmitting at 5 GHz and at speeds up to 54 Mbps using orthogonal frequency division multiplexing (OFDM) modulation technology. The 802.11b standard, now popularly known as Wi-Fi, specified operation in the 2.4 GHz band (also known as the ISM band) and could achieve speeds up to 11 Mbps using direct sequence spread spectrum (DSSS) technology. Because DSSS is easier to implement than OFDM, 802.11b products appeared on the market first, starting in late 1999. Since then, 802.11b products have been widely deployed in corporations, small offices/home offices (SOHO) and in residential home and in public locations (Wi-Fi "hotspots"). Products bearing the Wi-Fi logo conform to the 802.11b standard have passed an interoperability certification test defined by the Wireless Ethernet Compatibility Alliance (WECA) and have received permission from WECA to use the logo [11, 12].

In early 2001, the FCC announced new rules allowing additional modulations in the 2.4GHz range. This allowed IEEE to extend 802.11b to support higher data rates, resulting in the 802.11g standard, which is now in deployment stage. 802.11g defines new data rate, up to 54 Mbps, at 2.4 GHz using OFDM, while at the same time providing backward compatibility with 802.11b at speeds up to 11 Mbps using DSSS.

To transmit data over radio waves, WLAN devices must superimpose the data being transmitted onto the radio wave, also known as a carrier wave because it carries data. This process is called modulation. Different modulation types exist and each has its benefits and tradeoffs in terms of efficiency and power requirements. DSSS modulations are used in 802.11b/g. OFDM modulations are used in 802.11a/g. Together the frequencies of operation and the modulation types define the Physical Layer (PHY) of the IEEE standard. Products are compatible at the PHY layer when they use the same frequencies and modulation. A second data layer, the Medium Access Control Layer (MAC) has been standardized across 802.11a, b and g. Table 2.1 summarizes popular WLAN standards.

Table 2.1: Summary of 80.11 Standards

802.11		802.11a	802.11b	802.11g
Standard Approved	July 1997	September 1999	September 1999	September 2002
Available Bandwidth	83.5MHz	300MHz	83.5 MHz	83.5 MHz
Unlicensed Frequencies of Operation	2.4-2.48335GHz DSSS, FHSS	5.15-5.35GHz OFDM 5.725-5.825GHz OFDM	2.4-2.4835GHz DSSS	2.4-2.4835GHz DSSS, OFDM
Number of Non-Overlapping Channels	3(Indoor/Outdoor)	4 Indoor(UNII 1) 4 Indoor/Outdoor(UNII 2) 4 Outdoor (UNII 3)	3 (Indoor/Outdoor)	3 (Indoor/Outdoor)
Data Rate per Channel	2.1Mbps	54, 48, 36, 18, 12,9, 6 Mbps	11, 5.5, 2, 1 Mbps	54, 36, 33, 24, 22, 12, 11, 9, 6, 5.5, 2 1 Mbps
Modulation Type	DQPSK (2 Mbps DSSS) DBPSK (1 Mbps DSSS) 4GFSK (2Mbps FHSS) 2GFSK (1Mbps FHSS)	BPSK (6, 9 Mbps) QPSK (12, 18 Mbps) 16-QAM (24, 36 Mbps) 64-QAM (48, 54 Mbps)	DQPSK/CCK (11, 5.5 Mbps) DQPSK (2 Mbps) DBPSK (1 Mbps)	OFDM/CCK (6.9, 12, 18, 24, 36, 48, 54) OFDM (6, 9, 19, 18, 24, 36, 48, 54) DQPSK/CCK (22, 33, 11, 5.5 Mbps) DQPSK (2Mbps) DBPSK (1 Mbps)
Compatibility	802.11	Wi-Fi5	Wi-Fi	Wi-Fi at 11 Mbps And below

2.16 Carrier Sense Multiple Access with Collision Avoidance (CSMA/CA)

In an Ethernet LAN (IEEE 802.3), the Carrier Sense Multiple Access with Collision Detection (CSMA/CD) protocol establishes how packet collisions will be handled. In a WLAN, collision detection in this manner is not possible due to what is known as the "near/far" problem: in order to detect a collision, a station must be able to transmit and listen at the same time. To account for this difference, 802.11 use a slightly different protocol known as Carrier Sense Multiple Access with Collision Avoidance (CSMA/CA) or the Distributed Coordination Function (DCF).

CSMA/CA makes attempts to avoid packet collisions by using explicit packet acknowledgement (ACK), which means that an ACK packet is sent by the receiving station to confirm that a packet arrived intact. CSMA/CA works by having the station that wishes to transmit sense the air and if there is no activity detected, the station will wait an additional random period of time and if there still is no activity, it will transmit the data. If the packet is received intact, the receiving station will send an ACK frame and once it is received by the original sender the transmission is considered to be complete. If the ACK command is not received in a specified random period of time, the data packet will be resent, assuming that the original packet experienced a collision and did not reach the destination. CSMA/CA will also handle other interference and radio wave related problems effectively, but creates considerable overhead. Thus, a LAN with 802.11 traffic will always have slower performance than an equivalent Ethernet LAN [1, 11].

2.17 WLAN security [20]

Security is one of the most important features when using a wireless network and it is one of the biggest strengths for cellular wireless networks (WWANs) and one of the biggest weaknesses in 802.11 networks (WLANs). 802.11b networks have several layers of security, however there are weaknesses in all of these security features. The first level of security is to have wireless LAN authentication done using the wireless adapter's hardware (MAC) address. There is a table of MAC addresses programmed into the Access Points enabling the Access Points to reject access from any MAC address that is not present in the list. This process is called MAC address association. However, this alone is not secure because the MAC address of a wireless client can easily be falsely created.

Security can be increased on wireless LANs by using shared key authentication. This shared key must be delivered through a secure method other than the 802.11 connection. In practice, this key is manually configured on the Access Point and client, which is not efficient on a large network with many users. This shared key authentication is not considered secure and is not recommended to ensure security.

Another weakness in an 802.11 network is the difficulty in restricting physical access to the network, because anyone within range of a wireless Access Point can send, receive, or intercept frames. Wired Equivalency Protocol (WEP) was designed to provide security equivalent to a wired network by encrypting the data sent between a wireless client and an Access Point. However, key management is a significant problem with WEP. WEP keys must be distributed via a secure channel other than 802.11. The key is normally a text string that needs to be manually configured on the wireless Access Point and wireless clients, which is not practical in a large network. There is also no mechanism to change the WEP key regularly or periodically, so all wireless Access Points and clients use the same manually configured WEP. With several wireless clients sending large amounts of data, without changing the WEP key, it is possible to intercept data traffic and determine the WEP key. This would allow a hacker to intercept and decrypt the data traffic.

Another problem that has been reported with wireless LANs is that when the security features are turned on, there are problems with interoperability between wireless LAN modules from one vendor and wireless LAN Access Points from another vendor. Wireless LANs were designed specifically to operate in the 2.4 GHz band, which is a globally allocated frequency for unlicensed operation. This means that there is no requirement to be a licensed operator to run a wireless LAN in this frequency. A wireless WAN however operates in tightly regulated frequency spectrums and all operators must be licensed to operate in this frequency. This implies much better data security and protection, since licensed operators have to follow government regulations for wireless access. In contrast to the security weaknesses in 802.11 networks, cellular wireless WAN networks are extremely secure. These networks incorporate military technology and sophisticated encryption and authentication methods.

2.18 WLAN range / coverage

The distance over which RF waves can communicate is a function of product design and (including transmitted power and receiver design) and the propagation path, especially in indoor environments. Interactions with typical building objects, including walls, metal, and even people, can affect how energy propagates, and thus what range and coverage a particular system achieves. Most wireless LAN systems use RF because radio waves can penetrate many indoor walls and surfaces. The range (or radius of coverage) for typical WLAN systems varies from under 20 meters to more than 500

meters. Coverage can be extended through the use of microcells with each cell having an Access Point [14].

2.19 WLAN throughput [20]

As with wired LAN systems, actual throughput in wireless LANs is product and configuration dependent. Factors that affect throughput include airwave congestion (number of users), propagation factors such as range and multipath, the type of WLAN system used, as well as the latency and bottlenecks on the wired portions of the WLAN. Typical data rates range from 1 to 11 Mbps.

2.20 Integrity and reliability

Wireless data technologies have been proven through more than fifty years of wireless application in both commercial and military systems. While radio interference can cause degradation in throughput, such interference is rare in the workplace. Robust designs of proven WLAN technology and the limited distance over which signals travel result in connections that are far more robust than cellular phone connections and provide data integrity performance equal to or better than wired networking [13, 20].

2.21 Safety

The output power of wireless LAN systems is very low, much less than that of a hand-held cellular phone. Since radio waves fade rapidly over distance, very little exposure to RF energy is provided to those in the area of a wireless LAN system. Wireless LANs must meet stringent government and industry regulations for safety. No adverse health affects have been attributed to wireless LANs [14].

2.22 WLAN product certification

The Wireless Ethernet Compatibility Alliance (WECA) is the industry organization that certifies 802.11 products that are deemed to meet a base standard of interoperability. The first families of products to be certified by WECA are that based on the 802.11b standard. These products are

stamped with the Wi-Fi logo and referred to as Wi-Fi devices. 802.11a products will be stamped with the Wi-Fi5 logo. The Wi-Fi logos certify that the product will work with any other Wi-Fi certified device, regardless of manufacturer [13].

2.23 Chapter summary

Chapter 2 concerned itself with the introduction of the GPRS and 802.11b standards. Overview of basic functions, operations and architecture of both standards were discussed. The architecture discussion for networks included additional components for GPRS network and explained their functions. It was also given that WLAN has two architectures; either ad hoc or infrastructure. This chapter further discussed security issues of both networks and gave a comparison of the two networks in terms of coverage and throughput. Chapter 2 thus served as a background study for the research before GPRS/802.11b handover could be designed and implemented. Chapter 3 that follows discusses mobility between the two networks using the Mobile IP standard.

3. Mobile IP principles

3.1 Introduction

Mobile computing and networking are two different things. In mobile networking, computing activities are not disrupted when the user changes the computer's point of attachment to the Internet. Instead, all the needed reconnection occurs automatically and noninteractively. The most fundamental challenge in mobile networking is the way the Internet Protocol routes packets to their destinations according to IP addresses. These addresses are associated with a fixed network location. When the packet's destination is a Mobile Node, this means that each new point of attachment made by the node is associated with a new network number and, hence, a new IP address, making mobility impossible. The Mobile IP (RFC2002) was designed solely to solve this problem. Before going into deeper discussion we first have to know some of the important definitions that comprise Mobile IP as drafted by RFC 3220 and they are [15].

- The Mobile Node

  This is a host or router that changes its point of attachment from one network or subnetwork to another. A Mobile Node may change its location without changing its IP address; it may continue to communicate with other Internet nodes at any location using its (constant) IP address, assuming link-layer connectivity to a point of attachment is available.

- Home Address

  An IP address that is assigned for an extended period of time to a Mobile Node. It remains unchanged regardless of where the node is attached to the Internet.

- Home Network

  A network, possibly virtual, having a network prefix matching that of a Mobile Node's Home Address. Note that standard IP routing mechanisms will deliver datagrams destined to a Mobile Node's Home Address to the Mobile Node's Home Network.

- The Home Agent

  A router on a Mobile Node's Home Network which tunnels datagrams for delivery to the Mobile Node when it is away from home, and maintains current location information for the Mobile Node.

- The Foreign Agent

  A router on a Mobile Node's visited network, which provides routing services to the Mobile Node while, registered. The Foreign Agent detunnels and delivers datagrams to the Mobile Node that were tunneled by the Mobile Node's Home Agent. For datagrams sent by a Mobile Node, the Foreign Agent may serve as a default router for registered Mobile Nodes.

- Care-of-Address

  The termination point of a tunnel towards a Mobile Node, for datagrams forwarded to the Mobile Node while it is away from home. The protocol can use two different types of care-of address: a "Foreign Agent care-of address" is an address of a Foreign Agent with which the Mobile Node is registered, and a "co-located care-of address" is an externally obtained local address which the Mobile Node has associated with one of its own network interfaces.

- Correspondent Node

  A peer with which a Mobile Node is communicating. A Correspondent Node may be either mobile or stationary.

- Mobility Agent

  A generic term for a Home Agent or a Foreign Agent.

- Mobility Binding

  The association of a Home Address with a care-of-address, along with the remaining lifetime of that association.

- Tunnel

  The path followed by a datagram while it is encapsulated. The model is that, while it is encapsulated, a datagram is routed to a knowledgeable decapsulating agent, which decapsulates the datagram and then correctly delivers it to its ultimate destination.

3.2 How Mobile IP works

The Mobile IP protocol allows the MNs to retain their IP addresses regardless of their point of attachment to the network. This can be fulfilled by allowing the MN to use two IP addresses. The first one, called the Home Address, is static and is mainly used to identify higher layer connections, e.g., TCP. The second IP address that can be used by a MN is the Care-of Address (COA). While the Mobile Node is roaming among different networks, the Care-of Address changes. The reason of this is that the Care-of Address has to identify the mobile's new point of attachment with respect to the network topology. The home IP address assigned to the Mobile Node makes it logically appear as if the Mobile Node is attached to its Home Network. In Mobile IPv4 the Care-of Address management is achieved by an entity called Foreign Agent, but can also be administered by MN itself [15]. For the Correspondent Node, the Mobile Node seems to be attached to the Home Network independently of which Foreign Network it is currently visiting.

The Internet Protocol (IP) routes packets from a source endpoint to a destination by allowing routers to forward packets from incoming network interfaces to outbound interfaces according to routing tables. The routing tables typically maintain the next-hop (outbound interface) information for each destination IP address, according to the number of networks to which that IP address is connected. The network number is derived from the IP address by masking off some of the low-order bits. Thus, the IP address typically carries with it information that specifies the IP node's point of attachment [27].

The MN must keep its IP address the same to maintain existing transport-layer connections as it moves from one place to the other. In TCP, connections are indexed by a quadruplet that contains the IP addresses and port numbers of both connection endpoints. There is disruption and loss in connection if any of these four numbers are changed. On the other hand, correct delivery of packets to the Mobile Node's current point of attachment depends on the network number contained within the Mobile

Node's IP address, which changes at new points of attachment. To change the routing requires a new IP address associated with the new point of attachment.

3.3 MIP architecture

Mobility by use of MIP can be achieved in two major configurations of IP addresses, the FA care of address and the co-located COA configurations. The FA COA configuration makes use of Foreign Agent (FA) whereas the co-located COA does not make use of FA. In the latter the MN handles some of the functions of the FA e.g. decapsulation. The latter configuration is the one being exploited in this project since it reduces both costs and complexity of achieving MIP handover. The following section will explain both configurations starting with FA COA architecture.

3.3.1 FA COA architecture

The preceding section has pointed that disruption in one of the quadruplet numbers that assure continuous connectivity between the MN and the Correspondent Node (CN) results in disconnection between the two nodes. The MN acquires one of the COAs supplied by FA in the Foreign Network and registers this new address with its HA located in the Home Network via the FA. The MN is able to discover the FA COA from the broadcasted Agent Advertisements messages from FA. Discovery of FA COA is discussed in *section 3.5*. They contain among other things COAs, their lifetimes and services that are offered by the Mobility Agents. The FA COA does not necessarily have to be the IP address of the FA but it can be one of the addresses offered by FA. The scenario is depicted in *Figure 3.1*, which shows a Mobile Node that has changed its point of attachment, moving from network A (HN) to network B (FN).

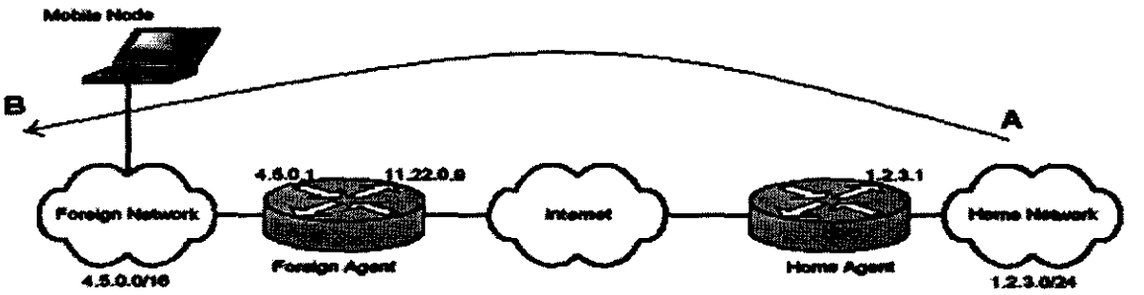

Figure 3.1 MN roaming

After obtaining a FA COA from the FA's Advertisement Messages, the MN registers this COA with its HA via the FA by issuing a registration request. After authentication of the MN to use its services, the HA replies to the FA to grant or deny the use of the HA services. By this time the mobility binding would have been created, which maps the HN address of the MN to its COA. The registration mechanism is discussed in *section 3.6.*

Figure 3.2 shows the scenario whereby the MN roamed from its HN to the Foreign Network whilst communicating with the Correspondent Node. A mobile agent (Home Agent) that is provided in a Home Network then receives traffic directed to the MN's home IP address even when the mobile client is not physically attached to the Home Network. In this way, the HA also functions as the gateway to the HN. This was imperative in the configuration of the testbed. When the Mobile Node is attached to a Foreign Network, a HA intercepts all packets destined for the MN and routes (tunnels) that traffic to a Foreign Agent using the mobile client's current care-of address.

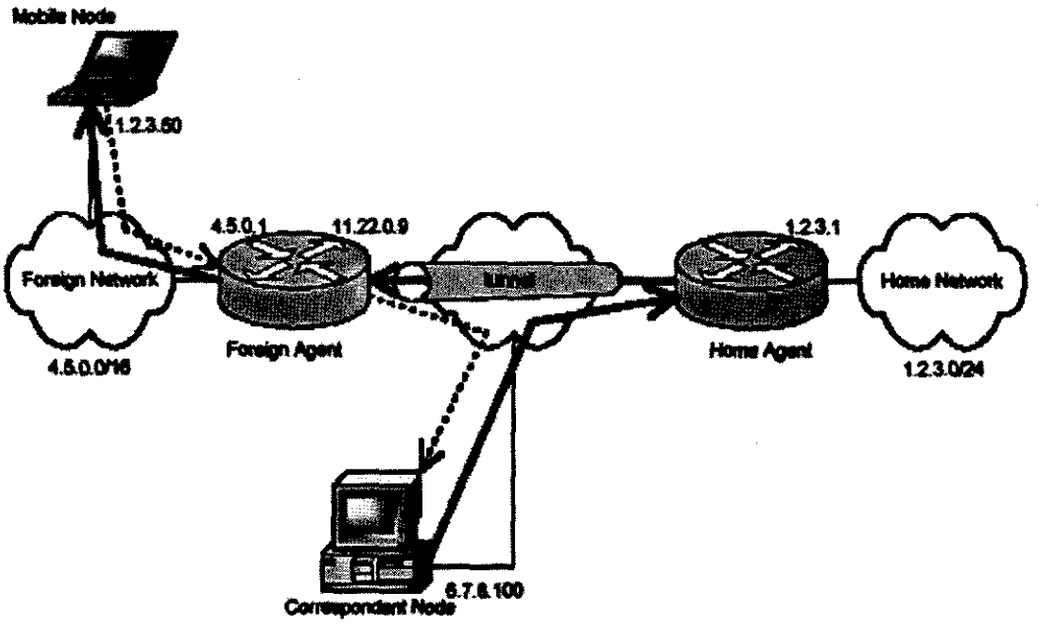

Figure 3.2 CN-MN route paths

It should be noted that the tunnel starts at HA and ends up at FA. The care-of address, which identifies the mobile client's current, topological point of attachment (the Foreign Agent) to the Internet, is used by the Home Agent to route packets to the Mobile Node. The FA decapsulates the received packets, and routes them to the MN. The process of encapsulation and decapsulation of packets is discussed in *section 3.4*. When the packets arrive at the MN, addressed to its Home Address, they will be processed properly by TCP or whatever higher level protocol that logically receives it from the Mobile Node's IP (that is, layer 3) processing layer. If the Mobile Node is not attached to a Foreign Network, the Home Agent simply arranges to have the packet data traffic delivered to the mobile client's point of attachment in the Home Network. Whenever the Mobile Node moves its point of attachment, it registers a new care-of address with its Home Agent [16, 32].

3.3.2 IP tunneling

In Mobile IP the HA redirects packets from the Home Network to the care-of address by constructing a new IP header that contains the Mobile Node's care-of address as the destination IP address. The new header then shields or encapsulates the original packet, shown in *Figure 3.3* causing the Mobile Node's Home Address to have no effect on the encapsulated packet's routing until it arrives at the care-of address. Such *encapsulation* is also called IP-within-IP encapsulation a mechanism specified in RFC2003 [33]. This suggests that the packet burrows through the Internet, bypassing the usual effects of IP routing. The tunnel is also designated by the thick pipe in the figure. The encapsulator can be taken as the HA and decapsulator as the FA with reference to the FA COA scenario discussed in the preceding section.

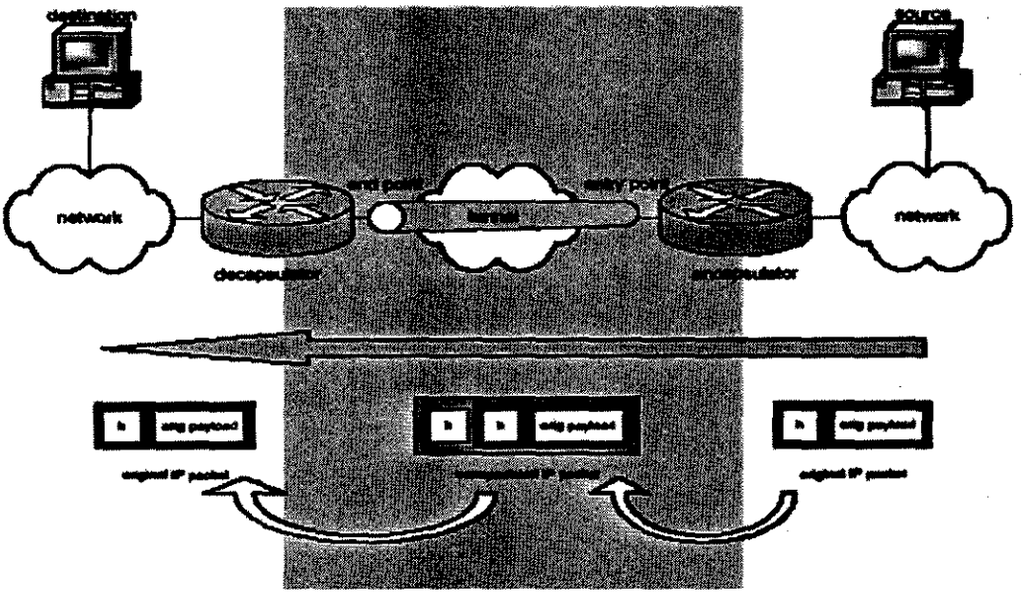

Figure 3.3 IP within IP encapsulation

When the packet arrives at the Foreign Agent the new IP header is removed and the original packet is sent to the Mobile Node for proper processing by whatever higher level protocol (layer 4) that logically receives it from the Mobile Node's IP (layer 3) processing layer. Another encapsulation mechanism is GRE, Generic Routing Encapsulation as specified in RFC2784 [33, 34].

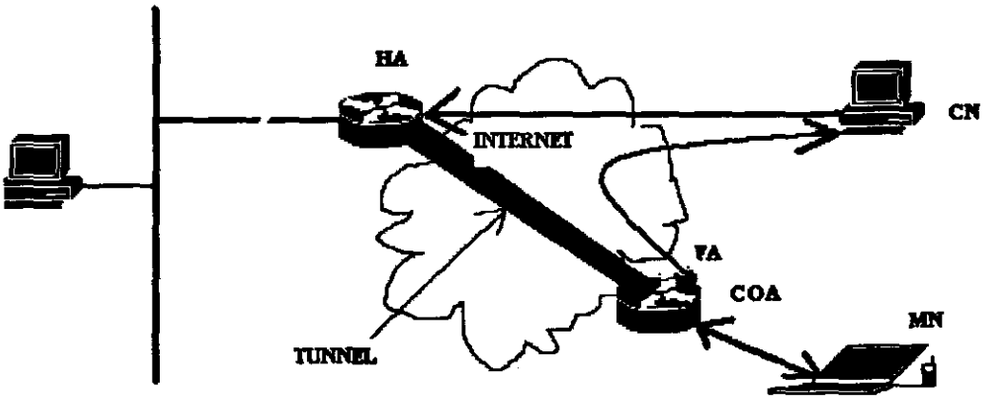

Figure 3.4 MIP Triangular routing

In the other direction, from the Mobile Node to the Correspondent Node, there is not necessarily a need for tunneling. In the basic operation, packets to the Correspondent Node are sent from the Mobile Node to the Foreign Agent. Since the Correspondent Node (in a basic scenario) is supposed to have a public routable address, it is possible for the Foreign Agent to directly forward the packet to the Correspondent Node. This mechanism of routing is called triangular routing as can be seen in the formation of arrows by the way packets are routed in *Figure3.4.* In this way the Home Agent is completely bypassed for Correspondent Node directed traffic. However, this data path is topologically incorrect because it does not reflect the true IP network source for the data, rather it reflects the Home Network of the Mobile Node. Because the packets show the Home Network as their source inside a Foreign Network, an access control list on routers in the network called ingress filtering drops the packets instead of forwarding them. A feature called reverse tunneling solves this problem by having the Foreign Agent tunnel packets back to the Home Agent when it receives them from the Mobile Node. This solution is discussed further in the next chapter under MIP issues [36].

Essentially reverse tunneling means that in addition to the forward tunnel (from the Home Agent to the Foreign Agent), the Foreign Agent also tunnels packets, from the Mobile Node, back to the Home Agent instead of directly sending them to the Correspondent Node. This mechanism does not use triangular routing as shown in *Figure 3.5* [15].

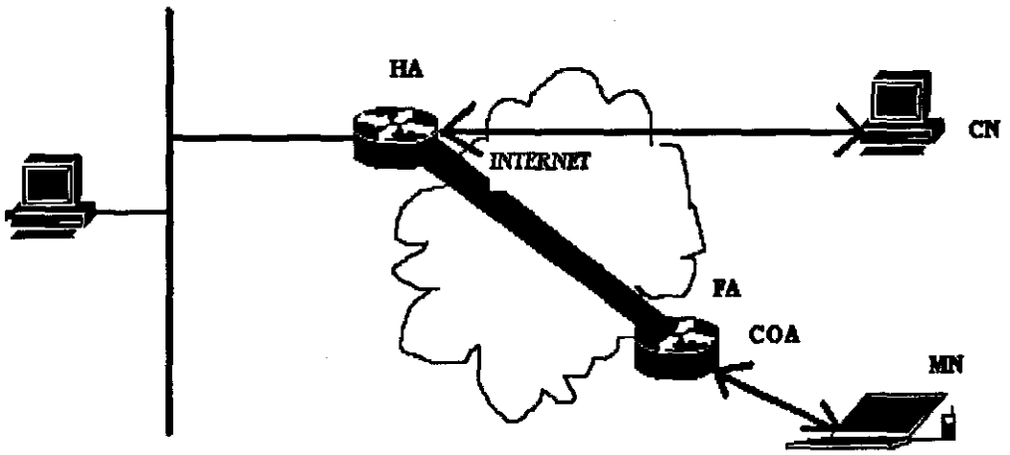

Figure 3.5 Reverse tunneling

3.3.3 Co-Located COA

The use of COA and Foreign Agent is effective but in order to reduce costs and complexity in the implementation of Mobile IP the co-located COA can be used without the FA. The Foreign Network does not have an FA installed, so here the MN will not receive any FA Advertisements. In this architecture, the MN can receive an IP address via DHCP or by dialup connection. The MN uses this address as the COA. Because it is assigned to an interface of the MN itself, it is called a co-located Care-of Address in contrast to FA COA. The MN registers itself directly with its HA in order to create a mobility binding. The HA can now set up a tunnel to the COA, which actually is the MN in this case. This situation is depicted in *Figure 3.6*.This architecture depicts that the MN responds to the CN by making use of normal routing.

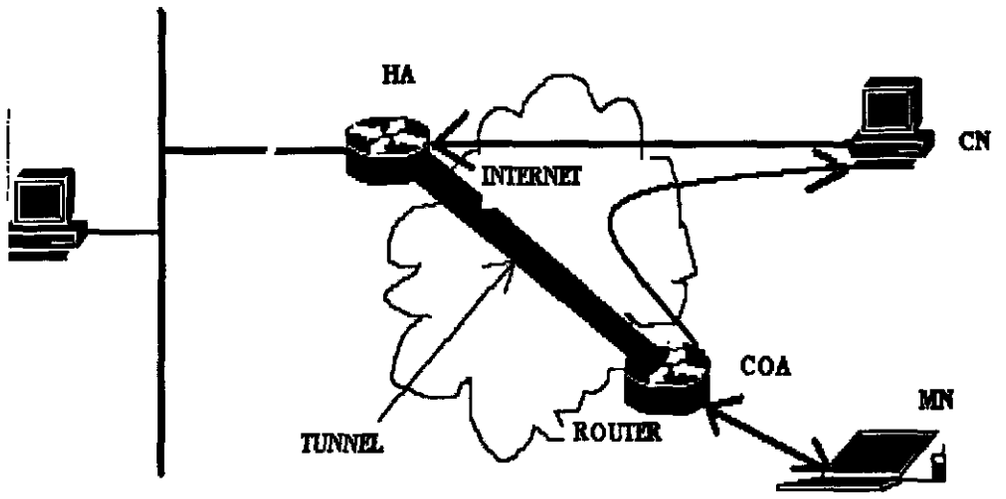

Figure 3.6 Co-located COA architecture

The packets destined for the Correspondent Node take the normal Internet path, but since this causes problems with most networks that use ingress filters the reverse tunneling mechanism should be used. Thus *Figure 3.7* shows that packets move from CN to MN via HA and go through the HA back to CN.

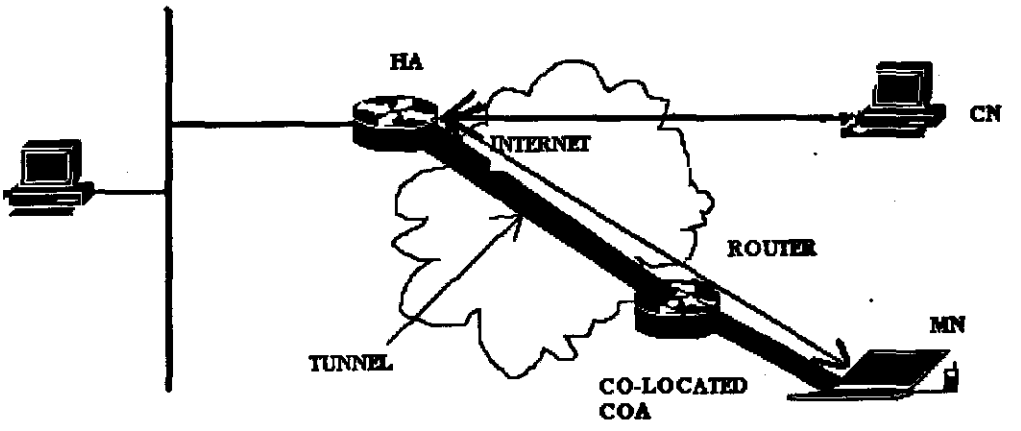

Figure 3.7 Reverse tunneling in co-located COA

Mobile IP, then, is best understood as the cooperation of three separable mechanisms:

- Discovering the care-of address;
- Registering the care-of address;
- Tunneling to the care-of address.

3.4 Discovering the COA

The Mobile IP *discovery* process has been built on top of an existing standard protocol, Router Advertisement, specified in RFC 1256 [18]. It is worth noting that Mobile IP discovery does not modify the original fields of existing router advertisements but simply extends them to associate mobility functions. Thus, a router advertisement can carry information about default routers, just as before, and in addition carry further information about one or more care-of addresses. When the router advertisements are extended to also contain the needed care-of address, they are known as *agent advertisements*.

Home Agents and Foreign Agents typically broadcast agent advertisements at regular intervals (for example, once a second or once every few seconds). If a Mobile Node needs to get a care-of address and does not wish to wait for the periodic advertisement, the Mobile Node can broadcast or multicast a solicitation that will be answered by any Foreign Agent or Home Agent that receives it. Furthermore

HAs use agent advertisements to make themselves known, even if they do not offer any care-of addresses. Thus, an agent advertisement performs the following functions:

- allows for the detection of mobility agents;
- lists one or more available care-of addresses;
- informs the Mobile Node about special features provided by Foreign Agents, for example, alternative encapsulation techniques;
- lets Mobile Nodes determine the network number and status of their link to the Internet; and
- lets the Mobile Node know whether the agent is a Home Agent, a Foreign Agent, or both, and therefore whether it is on its Home Network or a Foreign Network.

If advertisements are no longer detectable from a Foreign Agent that previously had offered a care-of address to the Mobile Node, the Mobile Node should presume that Foreign Agent is no longer within range of the Mobile Node's network interface. In this situation, the Mobile Node should begin to hunt for a new care-of address, or possibly use a care-of address known from advertisements it is still receiving. The Mobile Node may choose to wait for another advertisement if it has not received any recently advertised care-of addresses, or it may send an agent solicitation [15].

3.5 Registering COA

The Mobile Node is configured with the IP address and mobility security association (which includes the shared key) of its Home Agent. In addition, the Mobile Node is configured with either its home IP address, or another user identifier, such as a Network Access Identifier [38]. The Mobile Node uses this information along with the information that it learns from the Foreign Agent advertisements to form a Mobile IP registration request. It adds the registration request to its pending list and sends the registration request to its Home Agent either through the Foreign Agent or directly if it is using a co-located care-of address and is not required to register through the Foreign Agent as shown in *Figure 3.8* [19].

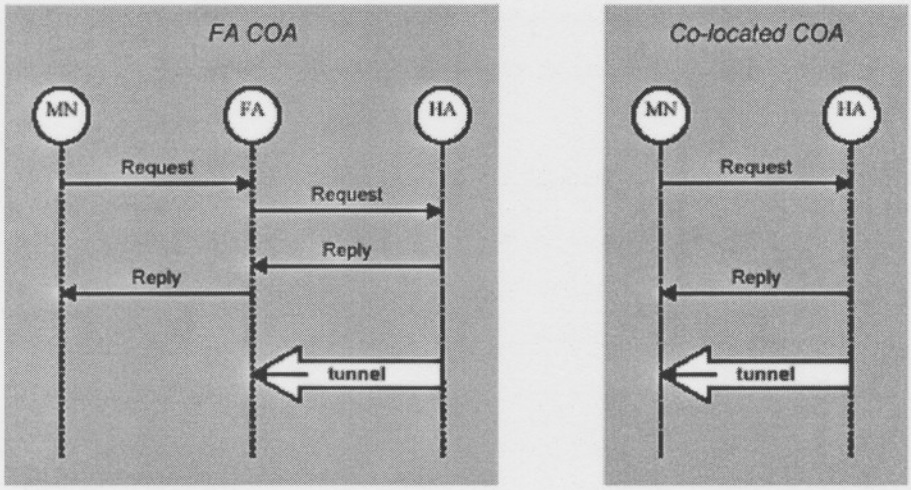

Figure 3.8 MIP Registration process

If the registration request is sent through the Foreign Agent, the Foreign Agent checks the validity of the registration request, which includes checking that the requested lifetime does not exceed its limitations, the requested tunnel encapsulation is available, and that reverse tunnel is supported. If the registration request is valid, the Foreign Agent adds the visiting Mobile Node to its pending list before relaying the request to the Home Agent. If the registration request is not valid, the Foreign Agent sends a registration reply with appropriate error code to the Mobile Node.

The Home Agent checks the validity of the registration request, which includes authentication of the Mobile Node. If the registration request is valid, the Home Agent creates a mobility binding (an association of the Mobile Node with its care-of address), a tunnel to the care-of address, and a routing entry for forwarding packets to the Home Address through the tunnel.

The Home Agent then sends a registration reply to the Mobile Node through the Foreign Agent (if the registration request was received via the Foreign Agent) or directly to the Mobile Node. If the registration request is not valid, the Home Agent rejects the request by sending a registration reply with an appropriate error code.

The Foreign Agent checks the validity of the registration reply, including ensuring that an associated registration request exists in its pending list. If the registration reply is valid, the Foreign Agent adds

the Mobile Node to its visitor list, establishes a tunnel to the Home Agent, and creates a routing entry for forwarding packets to the Home Address. It then relays the registration reply to the Mobile Node.

Finally, the Mobile Node checks the validity of the registration reply, which includes ensuring an associated request is in its pending list as well as proper authentication of the Home Agent. If the registration reply is not valid, the Mobile Node discards the reply. If a valid registration reply specifies that the registration is accepted, the Mobile Node is confirmed that the mobility agents are aware of its roaming. In the co-located care-of address case, it adds a tunnel to the Home Agent.

The Mobile Node reregisters before its registration lifetime expires. The Home Agent and Foreign Agent update their mobility binding and visitor entry, respectively, during re-registration. In the case where the registration is denied, the Mobile Node makes the necessary adjustments and attempts to register again. For example, if the registration is denied because of time mismatch and the Home Agent sends back its time stamp for synchronization, the Mobile Node adjusts the time stamp in future registration requests. Thus, a successful Mobile IP registration sets up the routing mechanism for transporting packets to and from the Mobile Node as it roams [27, 29, 32].

3.6 Authentication.

Registration requests contain parameters and flags that characterize the tunnel through which the HA will deliver packets to the care-of address. When a Home Agent accepts the request, it begins to associate the Home Address of the Mobile Node with the care-of address, and maintains this association until the *registration lifetime* expires. The triplet that contains the Home Address, care-of address, and registration lifetime is called a *binding* for the Mobile Node. A registration request can be considered a *binding update* sent by the Mobile Node. A binding update is an example of a *remote redirect*, because it is sent remotely to the Home Agent to affect the Home Agent's routing table. This view of registration makes the need for authentication very clear. The Home Agent must be certain registration was originated by the Mobile Node and not by some other malicious node pretending to be the Mobile Node [28].

A malicious node could cause the Home Agent to alter its routing table with erroneous care-of address information, and the Mobile Node would be unreachable to all incoming communications from the Internet. It is for this reason that each Mobile Node and Home Agent must share a security association and be able to use Message Digest 5 with 128-bit keys to create unforgeable digital signatures for registration requests. The signature is computed by performing MD5's one-way hash algorithm over all the data within the registration message header and the extensions that precede the signature [31].

To secure the registration request, each request must contain unique data so that two different registrations will in practical terms never have the same MD5 hash. Otherwise, the protocol would be susceptible to *replay attacks*, in which a malicious node could record valid registrations for later replay, effectively disrupting the ability of the Home Agent to tunnel to the current care-of address of the Mobile Node at that later time. To ensure this does not happen, Mobile IP includes within the registration message a special *identification field* that changes with every new registration. The exact semantics of the identification field depend on several details, which are described at greater length in the protocol specification [16]. Briefly, there are two main ways to make the identification field unique.

One is to use a timestamp; then eah new registration will have a later timestamp and thus differ from previous registrations. The other is to cause the identification to be a pseudorandom number; with enough bits of randomness, it is highly unlikely that two independently chosen values for the identification field will be the same. When randomness is used, Mobile IP defines a method that protects both the registration request and reply from replay, and calls for 32 bits of randomness in the identification field. If the Mobile Node and the Home Agent get too far out of synchronization for the use of timestamps, or if they lose track of the expected random numbers, the Home Agent will reject the registration request and include information to allow resynchronization within the reply. Using random numbers instead of timestamps avoids problems stemming from attacks on the NTP protocol that might cause the Mobile Node to lose time synchronization with the Home Agent or to issue authenticated registration requests for some future time that could be used by a malicious node to subvert a future registration.

The identification field is also used by the Foreign Agent to match pending registration requests to registration replies when they arrive at the Home Agent and to subsequently be able to relay the reply to the Mobile Node. The Foreign Agent also stores other information for pending registrations, including the Mobile Node's Home Address, the Mobile Node's Media Access Layer (MAC) address, the source port number for the registration request from the Mobile Node, the registration lifetime proposed by the Mobile Node, and the Home Agent's address. The Foreign Agent can limit registration lifetimes to a configurable value that it puts into its agent advertisements. The Home Agent can reduce the registration lifetime, which it includes as part of the registration reply, but it can never increase it.

As *Figure 3.4* shows, in Mobile IP Foreign Agents are mostly passive, relaying registration requests and replies back and forth between the Home Agent and the Mobile Node, doing mostly what they are told. The Foreign Agent also decapsulates traffic from the Home Agent and forwards it to the Mobile Node. The Foreign Agents note that Foreign Agents do not have to authenticate themselves to the Mobile Node or Home Agent. A bogus Foreign Agent could impersonate a real Foreign Agent simply by following protocol and offering agent advertisements to the Mobile Node. The bogus agent could, for instance then refuse to forward decapsulated packets to the Mobile Node when they were received. However, the result is no worse than if any node were tricked into using the wrong default router, which is possible using unauthenticated router advertisements as specified in RFC 1256 [18].

3.7 Automatic Home Agent discovery

When the Mobile Node cannot contact its Home Agent, Mobile IP has a mechanism that lets the Mobile Node try to register with another unknown Home Agent on its Home Network. This method of *automatic Home Agent discovery* works by using a broadcast IP address instead of the Home Agent's IP address as the target for the registration request. When the broadcast packet gets to the Home Network, other Home Agents on the network will send a rejection to the Mobile Node; however, their rejection notice will contain their address for the Mobile Node to use in a freshly attempted registration message. Note that the broadcast is not an Internet-wide broadcast, but a *directed* broadcast that reaches only IP nodes on the Home Network.

3.8 CHANGES WITH IP VERSION 6

IPv6 includes many features for streamlining mobility support that are missing in IP version 4 (current version), including Stateless Address Autoconfiguration and Neighbor Discovery. IPv6 also attempts to drastically simplify the process of renumbering, which could be critical to the future routability of the Internet. Since the number of mobile computers accessing the Internet will likely increase, efficient support for mobility will make a decisive difference in the Internet's future performance. This, along with the growing importance of the Internet and the Web, indicates the need to pay attention to supporting mobility. Mobility Support in IPv6, as proposed by the Mobile IP working group, follows the design for Mobile IPv4. It retains the ideas of a Home Network, Home Agent, and the use of encapsulation to deliver packets from the Home Network to the Mobile Node's current point of attachment. While discovery of a care-of address is still required, a Mobile Node can configure its care-of address by using Stateless Address Autoconfiguration and Neighbor Discovery. Thus, Foreign Agents are not required to support mobility in IPv6. IPv6-within-IPv6 tunneling is also already specified [20].

3.9 Mobile IPv6 vs. Mobile IPv4

The Mobile IPv6 uses the experiences gained from the design and development of Mobile IPv4 together with the new IPv6 protocol features. Mobile IPv6 shares many features with Mobile IPv4, but the protocol is now fully integrated into IPv6 and provides many improvements over Mobile IPv4. The major differences between Mobile IPv4 and Mobile IPv6 are [40]:

- Support for Route Optimization [40]. This feature is now built in as a fundamental part of the Mobile IPv6 protocol. In Mobile IPv4 the route optimization feature is being added on as an optional set of extensions that may not be supported by all IP nodes. However, this does not mean that always the route optimization option will be applied in Mobile IPv6. A MN may decide to use or not use this option.

- In Mobile IPv6 the functionality of the Foreign Agents can be accomplished by IPv6 enhanced features, such as Neighbour Discovery discussed in RFC1970 and Address Autoconfiguration discussed in RFC1971. Therefore, there is no need to deploy Foreign Agents in Mobile IPv6.

- Mobile IPv6 and IPv6 use the source routing feature. This feature makes it possible for a Correspondent Host to send packets to a Mobile Node while it is away from its Home Network using an IPv6 Routing header rather than IP encapsulation, whereas Mobile IPv4 must use encapsulation for all packets. However, in Mobile IPv6 the Home Agents are allowed to use encapsulation for tunneling. This is required, during the initiation phase of the binding update procedure.

- In Mobile IPv6 the packets which arrive at the Home Network and are destined for a Mobile Node that is away from home, are intercepted by the Mobile Node's Home Agent using IPv6 Neighbour Discovery rather than ARP as it is used in Mobile IPv4.

- In IPv6 a new routing procedure is defined and is called anycast. This feature is used in Mobile IPv6 for the dynamic Home Agent address discovery mechanism. This mechanism returns one single reply to the Mobile Node, rather than the corresponding Mobile IPv4 mechanism that used IPv4 directed broadcast and returned a separate reply from each Home Agent on the Mobile Node's home subnetwork. The Mobile IPv6 mechanism is more efficient and more reliable. This is due to the fact that only one packet need to be replied to the Mobile Node.

- All Mobile IPv6 control traffic can be piggybacked on any existing IPv6 packets. This can be accomplished by using the IPv6 destination options. In contrary, for Mobile IPv4 and its Route Optimization extensions, separate UDP packets were required for each control message.

3.10 Commercial Implementations

A growing part of Internet equipment vendors have implemented Mobile IP software in their products. Cisco Systems, supplier of most of Vodafone NL's IP equipment, has been supporting Mobile IP in Cisco IOSTM Software22 Releases 12.0(1) T and beyond. This means that any Cisco router can be configured as HA and/or FA. Other hardware vendors that implement MIP support are for example Lucent, Nortel, Motorola and 3Com. As Mobile IPv4 is not implemented in the IPv4 standard, simply because the mobility support was added long after the standard was set, a mobile device has to use

separate MIP software to be MIP enabled. For the PC, commercial MIP clients have been developed that work under MS Windows. Several MN client software packages are available at the time of writing this thesis. Table 3.1 gives results of Mobile IP products that are available. With these software products, a mobile device, like a notebook computer, can be always online using LAN, WLAN, and GPRS for example.

Table 3.1: Available Mobile IP client software

Vendor	Product	Website
Lifix	Go!	http://www.lifix.fi
Birdstep	Intelligent Mobile IP	http://www.birdstep.com
Greenpacket	SONaccess	http://www.greenpacket.com
ipUnplugged	Roaming Client	http://wwwipunplugged.com
Netseal	Mobile Private Network Client	http://www.netseal.com

3.11 Chapter summary

This chapter gave an overview of how Mobile IP works and its architecture. It also described the use both FA COA and co-located COA. The setup of a tunnel between FA and HA is described together with the setup of a tunnel between MN and HA. Registration process is discussed together with the use of the UDP port. A comparison and differences between Mobile IP version 4 and 6 was given. Chapter 4 that follows discusses handover solution for the demonstration setup.

4. Handover solutions

4.1 Introduction

The proliferation of 802.11b (Wi-Fi) based wireless LAN technology and the benefits arising from the associated broadband connectivity have given rise to an aggressive deployment of public hotspots that serve as broadband access networks. These hotspots are being deployed in prime locations like crowded restaurants, cafeterias, enterprises, university campuses, and airports. The connectivity services of these hotspots are primarily used to access and transfer data between a user's mobile terminal and her Home Network. The increasing number of such hotspots promises a higher bandwidth and connectivity on the move. Although 802.11 wireless LAN links boast Mbps bandwidth, their physical coverage is fundamentally limited because of the engineering constraints of the underlying radio technology. To increase the coverage, one can deploy multiple wireless LAN segments in an overlapped fashion as shown in *Figure 4.1* [1].

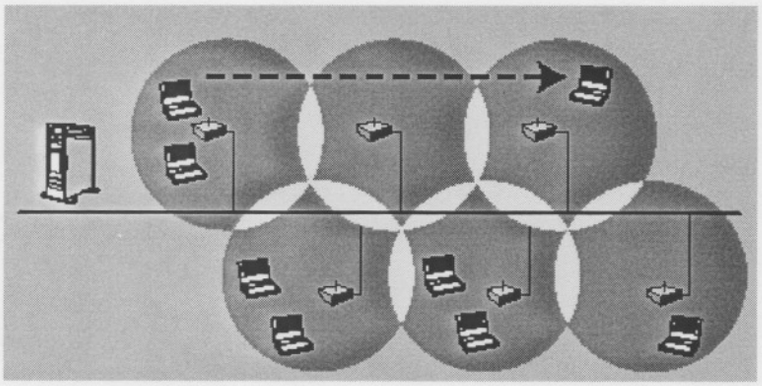

Figure 4.1 802.11b Overlapped scheme

As mobile terminals move across these overlapped segments, they can remain connected continuously by associating with appropriate Access Points based on the perceived radio signal strength and quality. The intelligence to measure the signal strength and switch among Access Points is built into the wireless LAN interface cards, which expose various status and control information to device drivers.

That is, existing 802.11 wireless LAN hardware supports link-layer handover without software involvement.

While the roaming capability provided by Mobile IP is good within and across the coverage areas of hotspots corresponding to foreign subnets, it is not adequate for retaining connectivity when one moves out of coverage area for durations long enough to cause already established sessions to timeout. Thus, Mobile IP support is effective for retaining session continuity as long as there is some physical medium available to communicate with the home subnet. If the users are moving across non-adjacent hotspots or if the users pass through "dead zones" in enterprise or campus networks, the connectivity is lost. The loss of connectivity for a prolonged period may lead to session timeouts, thus rendering the continuous network service of Mobile IP impossible.

Hence there is a need to use a WWAN that is capable of allowing roaming nodes to maintain data continuity between these APs and dead zones. This can be accomplished by making use of GPRS. GPRS was originally developed for GSM, but it is also being integrated in other cellular standards. GPRS is a new bearer service for GSM that improves and simplifies wireless access to packet data networks, e.g. Internet. It applies a packet radio principle to transfer user data packets in an efficient way between GSM Mobile Stations and external packet data networks. Packets can be directly routed from the GPRS Mobile Stations to packet switched networks.

GPRS uses the same radio resources (time slots) as GSM's voice service. Some portion of the radio resources are reserved for either of the services, data or voice. Hence, GPRS data rates are low as compared to 802.11b. GPRS networks have a larger coverage area of more than $1km^2$ per cell than pico-cells of few $100m^2$ offered by 802.11b. This implies that a GPRS user has more freedom of movement covering large areas, whereas in WLANs a user is confined to several meters in the range of the AP. Nevertheless, a user may want to access Internet at higher data rates than that offered by GPRS, thus may opt to use a nearby wireless hot-spot. The solution points to integration of both networks to allow the user to roam between them. The following table 4-1 summarizes differences between WLAN (802.11b) and WWAN (GPRS) [20].

Table 4.1: WLAN vs. WWAN comparison

	Wireless LAN	Wireless WAN
Coverage	Office Buildings or Campus with some public hotspots	Available wherever there is cellular network coverage; nationwide and global
Throughput Speeds	1-5 Mbps (However the underlying Internet connection may yield a slower speed)	30-50 kbps (GPRS) 40-70 kbps (CDMA2000 1X)
Security	Security flaws	Secure encryption and authentication
Airtime Charges	Airtime charges exist for most Hotspots access. No airtime charges for office or home users (although ISP monthly service fee still exists)	Monthly subscription from wireless network provider
Uses	Accessing a shared network within a building or across a campus	Remote access to a corporate network for e-mail and applications Web and Internet access
Voice	No	Yes
Wired analogy	Ethernet Network	Remote modem access
Advantages	• High speed • No airtime charged to set up networks (hardware costs and broadband Internet connection fee still apply	• Ubiquitous coverage • Secure Network • Access your data from anywhere
Disadvantages	Localized coverage only Security problems	Data rates faster than dial up, but not at wireless LAN speeds yet

One of the aims of the project is to deliver a demonstration of a Mobile IP implementation. The demonstration should allow a Mobile Node (MN) to be always connected to the Internet using Vodacom or MTN GPRS and a WLAN Access Point at the NWU. The MN, if required can also use

wired Ethernet connections. The goal of the project is to show handovers based on different signal strength criteria

4.2 Handover architecture

Handover is the mechanism by which an ongoing connection between a mobile terminal or host and a corresponded terminal or host is transferred from one point of access to another. In cellular voice telephony and mobile data networks, such points of attachments are referred to as base stations and in wireless LANs, they are referred as Access Points. In either case, such a point of attachment serves a coverage area called a cell. Handover, in the case of a cellular telephony, involves the transfer of voice call from one BS to another. In the case of WLANs, it involves transferring the connection from one AP to another. In hybrid networks it will involve the transfer of a connection from one BS to another BS from an AP to another AP, between a BS and an AP, or vice versa [4].

There can be many different scenarios to implement inter-technology roaming between GPRS and WLAN networks. *Figure 4.2* shows five different scenarios that enable the implementation of inter-technology roaming between WLAN and GPRS. Note that the numbers indicate each scenario's interaction between the networks.

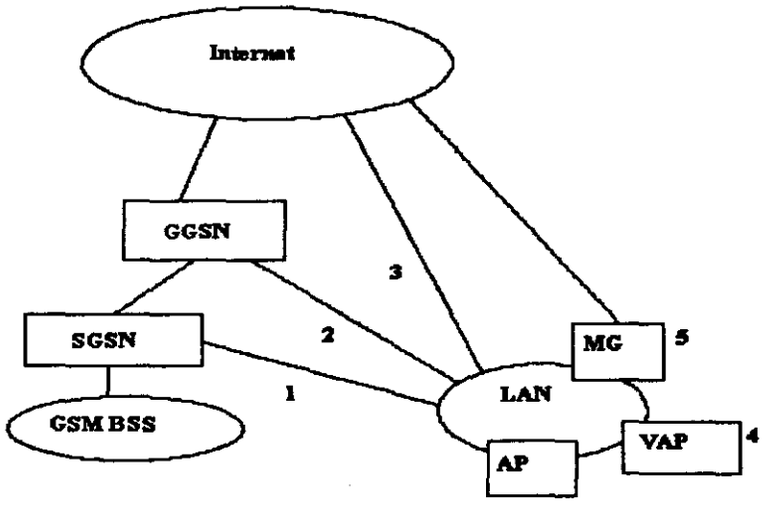

Figure 4.2 GPRS/WLAN architectures

In the first two architectures, GPRS is a master and WLAN is a slave network. The third scenario considers exploiting the Mobile IP for implementation of inter-technology roaming. The fourth scenario considers implementation of the GPRS network as an Access Point in the WLAN. The fifth scenario considers using a mobility gateway to interconnect the two networks. In the fourth scenario, WLAN is the master network and GPRS is a slave for the inter-technology roaming. In the third and fifth scenario, GPRS and WLAN are interconnected as peers in a larger network. It is preferable in this project to reduce, as far as possible, major changes to existing networks and technologies especially at the lower layers such as MAC and physical layers. This will ensure that existing networks will continue to function as before. Scenarios 3 and 5 were preferred with scenario 3 shown in *Figure 4.3* being implemented in this project to reduce the number of entities and complexity. This is because there are no changes to either GPRS or WLAN specific protocols [2].

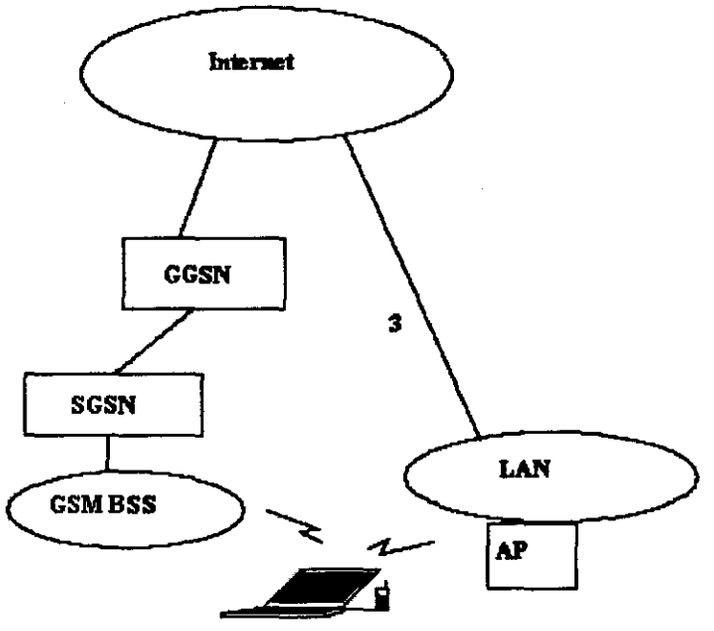

Figure 4.3 GPRS/8082.11b handover architectures

4.3 Initiating handover

There are various ways of initiating a handover in hybrid networks like GPRS and WLANs networks. The handover could be initiated by an entity in either of the networks or can be initiated by the MN

itself. The first option dictates that a network decides when it is suitable for the MN to change point of attachment or network. This requires modification in the master network and results in the unfavourable conditions of costs to modify the network. The second option suggests the self determination of the Mobile Node to take control of the suitable condition to make a handover. This is favourable as already indicated by *Figure 4.3*. The third scenario is used in this project and requires the decision making module in the MN to make favourable handovers.

4.4 Handover Metrics

There are different metrics that could be implemented to make a suitable handover. The Received Signal Strength (RSS) is the traditional metric that has been implemented in homogenous networks to make handovers. The RSS is implemented in this project to make handovers between hybrid networks like GPRS and 802.11b. The RSS in combination with other metrics was be used in this project to make a suitable handover. The combinational metrics include, dwell time and number of counts the current signal remains under the minimum threshold.

4.5 Mobile IP problems

It was mentioned in chapter 3 that the Mobile IP mechanism can be disrupted if not rendered not to work by problems found in the setup and securities of the access networks. The problems that were mentioned are the use of private IP addresses using a Network Address Translation (NAT) gateway and the other is the use of ingress filter by routers. Both issues are problems that can be resolved as in the following two sub-sections. The use of the NAT gateway is mostly found in the GPRS network as was the case in this project and the use of ingress filtering by routers is mostly found in LAN or WLAN.

4.5.1 NAT Problem in GPRS

There are several problems of with Mobile IP handovers with regard to the GPRS mobile network. The main problem is the Network Address Translation (NAT) which is used by the mobile network operator. The operator uses NAT to map the internal private IP addresses of the mobile subscribers to

public IP addresses to provide Internet services. Using private IP addresses by the subscribers is more efficient for the operator as IPv4 addresses are being saturated, but private IP addresses are not routable outside the provider's private network. This was the case with the mobile network setup used in this project. Therefore, the operator supports the Internet access by using a NAT Gateway which translates the private IP address of each Mobile Node to the gateway's own external public IP addresses. Due to the fact that Mobile IP functions are based on routable public IP addresses, Mobile IP services are not reachable for subscribers with private IP addresses [40].

The tunneling function of Mobile IP is incompatible with the Network Address Translation [23]. The reason for this is given by the Mobile Node behind the NAT Gateway which has been allocated a private IP address. Thus, the Mobile Node has no routable IP address and it is impossible to establish a link to the Home Agent. Such a link is required to set up the tunneling functions. Otherwise, Mobile IP will not work across the NAT Gateway. Furthermore, tunneling functions which are used by Mobile IP fail due to the NAT Gateway. Without these tunneling functions it is impossible to establish a Mobile IP based communication to the mobile user.

There are some implementations available to solve the NAT problems for Mobile IP. In [24] a solution is described which is based on the Mobile IP's messages which pass through a NAT Gateway. The messages pass the NAT Gateway by sending them as UDP packets on port 434 shown in *Figure 4.4.* A Mobile IP extension which sends all traffic (signaling and data) via UDP tunnel at UDP port 434 is defined.

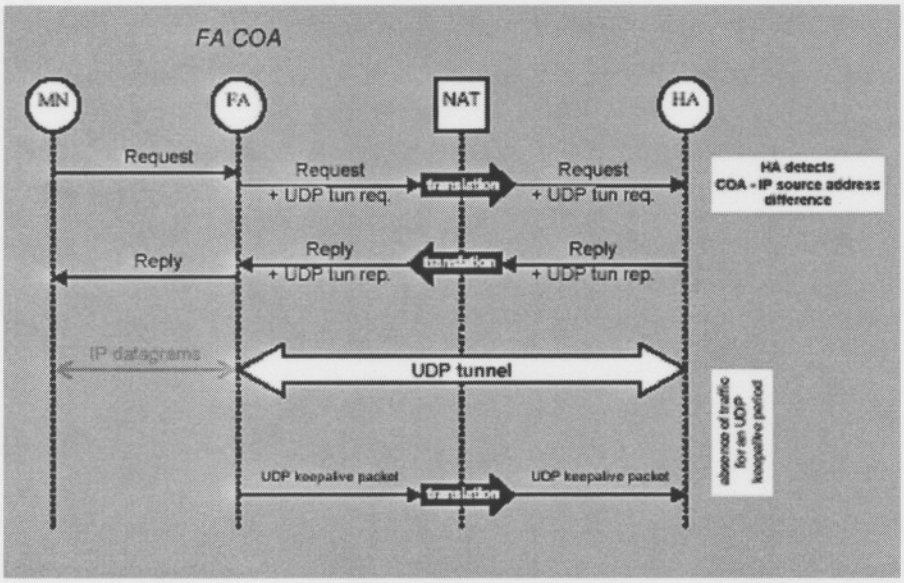

Figure 4.4 MIP UDP Tunneling through NAT [32]

This results in IP in UDP, GRE in UDP, and minimal encapsulation in UDP tunneling. To make sure the mapping in the NAT will not be deleted, a keepalive packet must be sent when no packets have been sent for a specific period, called the keepalive interval. It is very likely that reverse tunneling also will be needed. There are two new registration extensions defined for this purpose: the UDP Tunnel Request Extension and the UDP Tunnel Reply Extension.

In case of a co-located COA, the MN adds the UDP Tunnel Request Extension to its Registration message (before the MH authentication extension) to notify the HA that it is capable of handling MIP UDP tunneling. In case of an FA COA, the FA adds the UDP Tunnel Request Extension to its Registration message (before the FH authentication extension) to notify the HA that it is capable of handling MIP UDP tunneling; the MN does not know anything about the UDP tunneling then. In both cases the HA may detect that the MN/FA is behind a firewall because the COA differs from the IP source address. If the HA has detected this difference and supports MIP UDP Tunneling, it will reply with an MIP UDP Tunneling Reply Extension to notify the MN/FA that the it will use MIP UDP tunneling. The UDP tunnel extensions also exchange the tunneling mode to be used. *Figure 4.4* above shows a schematic view of this registration process in case an FA COA and reverse tunneling is used.

After the registration the tunnel is already set up because it uses the same UDP "session". In order to keep the UDP tunnel alive through the NAT (i.e. retain the address/port mapping), the FA sends a keepalive packet to the HA when no traffic has been sent for an UDP keepalive period [22, 38].

4.5.2 Mobile IP and Firewalls

Most of the mobile operators and ISPs have installed firewall systems to protect private networks from the Internet. Some firewall systems support ingress filtering. In this case, for Mobile IP, it is impossible to establish data connections through the firewall because the IP source addresses of the Mobile Node are not part of the local network's address space. Moreover, MIP requires a movement detection to detect movements between network systems. The GPRS network implements this by using additional Mobile IP functions [20].

The Mobile Node behind a firewall is sending data packets to a Correspondent Node with a different IP source address. The source address is the destination address of the Home Network and is different from the network address of the Mobile Node and the firewall supports ingress filtering. In this case, it is impossible for the Mobile Node to send data. In environments where this is a problem, Mobile Nodes may use reverse tunneling with the Foreign Agent supplied care-of address as the Source Address. Reverse tunneled packets will be able to pass normally through such routers, while ingress filtering rules will still be able to locate the true topological source of the packet in the same way as packets from non-Mobile Nodes [21].

4.6 Software

The Linux Operating System was chosen as a basis to build the MIP infrastructure for several reasons. Linux is used without licence fees and is available under GNU General Public Licence, which means that its source code is freely distributed and available to the public. This is an Operating System in which the user can fine-tune more than in other Operating Systems. It is a stable open source system, of which updates are well spread over the Internet. Hence Debian was chosen for the MIP infrastructure as this is one of the accessible Linux distributions. It is a package-based system with each set of packages containing the code for a specific piece of software. Together it forms the Debian

distribution, which is based on the Linux Kernel 2.4.18 or 2.4.20. The demonstration for this project is based on Debian version 3.0 with Linux Kernel 2.4.18.

The literature study indicated different available Mobile IP stacks that work with Linux. The choice was made to use HUT Dynamics v0.8.1 Mobile IP, because it is very well documented, there is an active mailing list, and it works under Redhat and Debian with Linux Kernel 2.4.18 or 2.4.20. HUT Dynamics v0.8.1 contains three important daemons, each for one of the MIP nodes: the MN daemon, the Foreign Agent (FA) daemon, and Home Agent (HA) daemon designated as *dynmnd, dynfad* and *dynhad* respectively. These daemons can be found on the Debian system in the directory */usr/local/sbin/,* and can be configured by a configuration (text) file containing all settings. These files are *dynmnd.conf., dynfad.conf* and *dynhad.conf* respectively and are located in */usr/local/etc.* Dynamics v0.8.1 software can be installed using Debian packages, which are specially created for Debian users. Additionally another option can be used; this is to download the entire source files and compiling them yourself. There are also API commands that can be given via executables *dynmnd_tool, dynfad_tool* and *dynhad_tool* also located in */usr/local/sbin/* while the daemons are running. These executables give status information of each corresponding daemon and thus MIP situation on request.

4.7 Chapter summary

Chapter 4 discussed most of the handover issues related to handover architecture and metrics. The NAT problems were emphasized together with firewall problems. The solution of sending registration messages through the NAT gateway is graphically explained. A comparison between WWAN and WLAN was given and differences explained. The use of HUT Dynamics v0.8.1 software and its location of important daemons: *dynhad, dynmnd* and *dynhad* are discussed. The following chapter discusses the GPRS/802.11b testbed.

5. Demo- Set-Up

A testbed of the GPRS/Wi-Fi handover that is based on different criteria of signal strength was set-up and is explained in the following sections. The testbed uses the Hut Dynamics Mobile IP software running on Debian Linux distribution Kernel 2.4.18. It will be explained how the important daemons; *dynmnd* and *dynhad* are configured. The involved configuration files for the daemons are given in Appendix B and C.

5.1. Set-up overview

A schematic of the demo set-up is depicted in *Figure 5.1*. The schematic shows nodes of the Foreign Network (FN) uncoloured, which are the WLAN interface on the MN, the WLAN Access Point (AP) and the internal interface of the HA interface. The AP is directly connected to the internal interface of the HA. This was to avoid broadcast messages from the EEI Ethernet in order to get clear graphical readings. It should be noted that no DHCP address allocation mechanism was used in the FN but private IP addresses as indicated by the 192.168.x.x prefix in the figure.

The IP address of the external interface of the HA is assigned dynamically by the DHCP mechanism of the Telkom ISP, hence the ppp0 designation. This address is assigned dynamically and it is a routable IP address that can be reached via two routes. The address is reachable by the WLAN interface via the Access Point and through the Home Agent's internal interface. It can also be reached by the GPRS interface through the GPRS network that connects to the Internet. This also applied to the connection establishment in the GPRS network.

5.1.1 Foreign Network

The WLAN network was chosen as the Foreign Network to avoid the problems of the ingress firewall and NAT gateway in the GPRS network. There is no ingress firewall between the MN and the HA. This implies that registration messages sent by the GPRS interface when entering

the FN will not be dropped. The registration messages would be dropped if there was a firewall because the IP address of the GPRS interface would have a different IP address as that allowed by the firewall to pass. The firewall would see the GPRS address as an external address initiating a session from within the WLAN network.

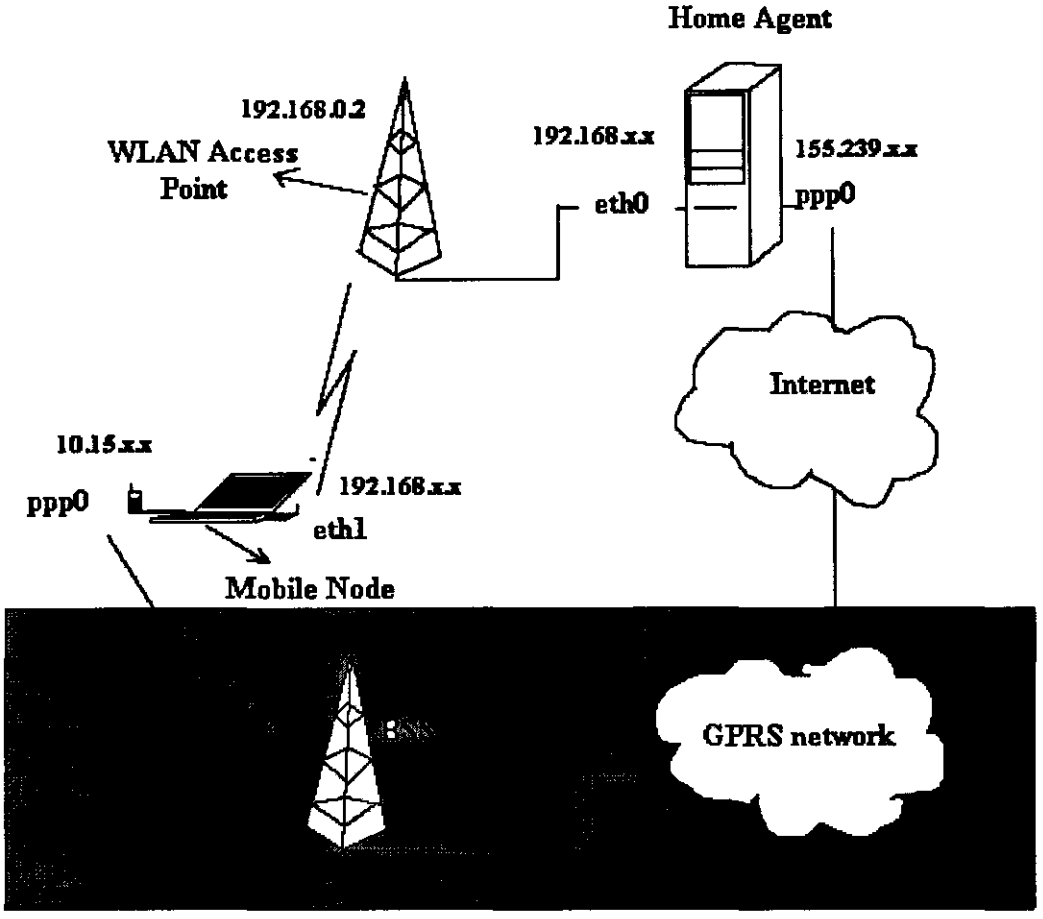

Figure 5.1 Demo set-up architecture

The Access Point is configured with its software to allow specific MAC addresses to use its services. In the demo-setup only the MAC address of the WLAN interface is allowed to use the services of the Access Point. This is a security measure implemented in the WLAN network to prevent other WLAN cards to reach the HA and is termed MAC address association with the Access Point. It should also be noted that the WLAN interface, Access Point and the internal

interface of the HA use IP addresses with prefix 192.168.x.x. All IP addresses with this prefix are regarded in the demo as the foreign IP addresses.

The WLAN interface is configured to use the architecture of an Infrastructure Basic Service Set and it is assigned a private IP address 192.168.0.3. The WLAN IP address will be used by the MN as its co-located COA in the FN. This configuration is performed in Linux files */etc/pcmcia/network.opts*, */etc/pcmcia/wireless.opts* and */etc/network/interfaces* found in Appendix E. The important settings in these files are as follows:

In the file */etc/pcmcia/network.opts*:

IPADDR	192.168.0.3
NETMASK	255.255.255.0
NETWORK	192.168.0.0
BROADCAST	192.168.255.255

In the file */etc/pcmcia/wireless.opts*:

ESSID	AP11B
MODE	managed
RATE	auto

In the file */etc/network/interfaces*:

```
iface eth1 inet static
        address     192.168.0.3
        netmask     255.255.0.0
        network     192.68.0.0
        broadcast   192.168.255.255
```

To keep the demonstration less complex, no encryption was used for data transfer between the Access Point and the WLAN interface. It should also be noted that the drivers for the WLAN card can either be downloaded or compiled in the Kernel or it can be activated during the Linux installation. The first option was used in the setup, since the updated versions of the drivers provide access to information on WLAN card's RSS, SNR and data link rate. The other reason is that the card could not work with the default driver of the Kernel.

The internal interface of the Home Agent is configured in Linux to use IP address 192.168.0.4 in the file */etc/network/interfaces* as explained in the preceding paragraph. The driver for the internal interface could also be activated during the Linux installation or downloaded from the Internet and compiled as a module in the Kernel. The latter option was used since the Kernel 2.4.18 had no driver for this integrated LAN interface, so the module was always activated when the service of the interface was needed. This interface also functioned as the gateway for the WLAN network to reach the external interface of the HA. The configuration was done as follows:

In the file */etc/network/interfaces*:

```
iface eth0 inet static
        address     192.168.0.4
        netmask     255.255.0.0
        network     192.68.0.0
        broadcast   192.168.255.255
```

5.1.2 Home Network

It was mentioned in the preceding chapters that the GPRS network uses the NAT gateway to map private IP addresses to the external routable IP addresses of the NAT gateway. The use of the NAT gateway could prevent the private IP address of the GPRS interface to register or deregister with the HA. Mobile IP is designed to avoid this problem by allowing the MN to use the private IP address and use UDP port 434 to register or deregister itself with the HA.

The GPRS interface uses IP address with prefix 10.15.x.x and this address is used as the Mobile Node's Home Address. The configurations of the GPRS interface were done by activating the PCMCIA options in the Kernel and editing the files *etc/chatscript/vodacomGPRS* and *etc/ppp/peers/vodacomGPRS* both given in Appendix D.

The external interface of the Home Agent is a modem of which the dial-up configurations were performed using software *pppconfig* that comes with Linux. The interface uses the Telkom ISP as means to connect to the Internet. It obtains a dynamic IP address every time it connects to the Internet and uses this IP address as the Home Agent's IP address. This IP address is reachable by both the WLAN card and the GPRS card via the WLAN and GPRS network respectively. The configuration script is created by using the command *pppconfig* and filling necessary attributes that apply to the setup of the Internet connection.

5.2 Mobile IP Daemons

Hut Dynamics software was downloaded and compiled in the Linux Kernel. It provides two important daemons for setting up the Mobile IP architecture needed in this project. The Mobile Node uses the daemon *dynmnd* which is configured by editing the *dynmnd.conf* file in the directory *usr/local/etc*. The Home Agent uses the daemon *dynhad* which can be configured using the file *dynhad.conf* in directory *usr/local/etc*. Both configuration files are given in Appendix B and C respectively. The network options for both daemons that are required during Linux installation or Kernel compilation are:

- Packet socket (CONFIG_PACKET)
- Kernel/User netlink socket (CONFIG_NETLINK)
- Routing messages (CONFIG_RTNETLINK)
- IP: Socket Filtering (CONFIG_FILTER)
- IP: tunneling (CONFIG_NET_IPIP)

5.2.1 Dynmnd Configuration

The *dynmnd.conf* provides settings of the Mobile Node's Home Address and the address of the Home Agent. The reverse tunneling mode is activated as opposed to the triangular tunneling, this done in order to avoid the NAT gateway in the GPRS network. The important configurations that had to be done to setup the required behaviour of the Mobile Node are:

In the */usr/local/etc/dynmnd.conf*:

```
# 1 = automatic, prefer reverse tunnel (bi-directional)
# 2 = automatic, prefer triangle tunnel (only CN->MN)
# 3 = accept only reverse tunnel
# 4 = accept only triangle tunnel
TunnelingMode < 1 >
```

It is seen here that the reverse tunneling is preferred to triangular tunneling. The Security Parameter Index (SPI) and Shared Secret between the MN and HA are set in the same file as follows:

```
# SPI < number >
SPI     1000
# SharedSecret < HEX number string or character string >
SharedSecret   "test"
```

These are some of the important configuration attributes with inclusion of the MN's and HA's IP addresses settings as follows:

```
MNHomeIPAddress 10.15.185.143
HAIPAddress 155.239.170.110
```

The behaviour of the MN is also influenced by configuration settings in the *dynhad.conf* of the HA as it will be realized in sub-section that will follow.

5.2.2 Dynhad Configuration

The Agent Advertisement (AA) messages were disabled so that the MN would not seek for the Foreign Agent. The HA was setup to support reverse tunneling, be accessible via two IP addresses and have security setup between HA and MN by making use of SPI and Shared Secret.

```
INTERFACES_BEGIN
# interface ha_disc agentadv interval force_IP_addr
    ppp0        1       1       10      155.239.170.110
    eth0        1       1       10      192.168.0.4
INTERFACES_END

# Interfaces to be used for Mobile IP services. Note that you have to configure
# each interface that may receive or send registration messages.
# interface: name of the interface, e.g. eth0
# ha_disc:
#   0 = do not allow dynamic HA discovery
#   1 = allow dynamic HA discovery with broadcast messages
# agentadv:
#   0 = do not send agent advertisements without agent solicitation
#   1 = send agent advertisements regularly
#  -1 = do not send any (even solicited) agent advertisements
# interval: number of seconds to wait between two agentadvs
# force_IP_addr: local address to be forced for this interface
#                (can be used to select one of the multiple virtual
#                addresses); if not entered, the primary address of the
#                interface is used
```

Some of the things that can be recognized in the configuration above are that the HA has two IP addresses and does not send any Agent Advertisement messages. For security configuration the following setup was performed:

```
AUTHORIZEDLIST_BEGIN
# SPI        IP
1000         10.15.185.143
AUTHORIZEDLIST_END

SECURITY_BEGIN
#       auth.   replay  timestamp     max         shared
# SPI   alg.    meth.   tolerance     lifetime    secret
1000    4       1       120           600         "test"
SECURITY_END
```

The configuration shows the association of the Mobile Node's home IP address with the SPI for authentication purposes and the use of shared secret "test" for security purposes.

5.3 Handover Script

This section gives the functions of the handover script that controls the Mobile IP mechanism. The script contains functions that control the routing table of the Mobile Node and with this routing information, the script is able to route packets in the correct route path to the correct outbound interfaces. The script reads the MN's IP configuration with the command *ip route*, which only shows interfaces that already have been assigned an IP address. The initializing phase of the script starts with the declaration of variable arrays for each interface and these arrays store information; ip address, subnet/ppp peer and gateway. A flowchart of the initializing phase of the script is given in *Figure 5.2*. Functions that are later used in the never ending loop are firstly declared and defined in the initializing phase.

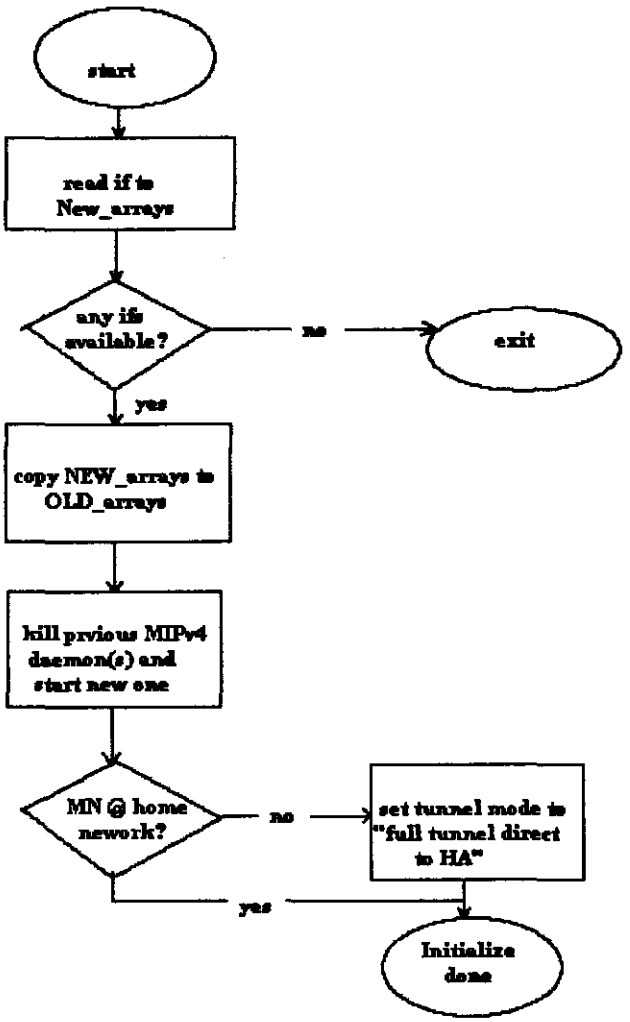

Figure 5.2 Initializing phase of handover script [32]

The flowchart of the initializing phase of the script shows that the New_arrays are first filled with ip address, gateways and network addresses. If the interfaces are available then this information can be copied to old arrays and a new daemon of the Mobile Node can be started. If it is discovered that there are no interfaces available the script will prompt for activation of these interfaces and exit. Then if the interfaces are available the script will check if the MN has the interface that is assigned the Home Address. If there it is found that there is no interface that has the Home Address, the script will setup a tunnel directly to the Home Agent.

A flowchart of the never-ending loop for the handover script is given in *Figure 5.3*. After the initializing phase the script has an option to use manual setting. Since the script is set to operate automatically the manual settings option is skipped.

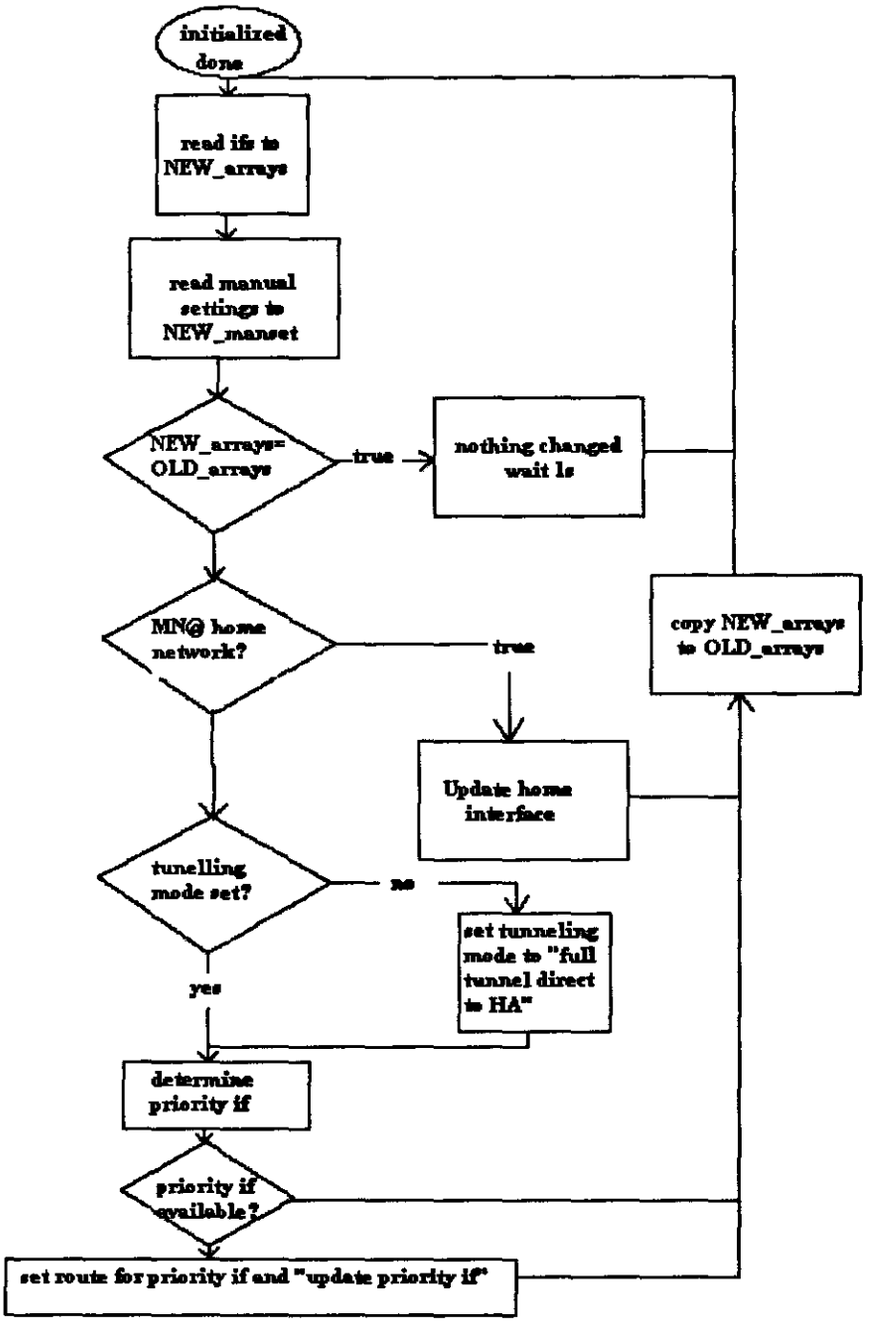

Figure 5.3 Never ending loop of handover script [32]

The script continues the execution by checking for any changes to the new arrays and old arrays. The changes are noticed if new arrays are not the same as old arrays, and the script will check if there are any interfaces that have the MN's Home Address. If false, the tunnel must be setup between the MN and the HA using the MN's preferred interface determined by the *PRIORITY_if* function. The *PRIORITY_if* function is the most important one since it determines the interface that can be used based on any criteria. The criteria being used to decide which interface to use is based on signal strength, but any criteria could be implemented in the *PRIORITY_if* function.

Some of the Mobile IP commands that are used in the handover script are:

- *dynmnd*: to start the Mobile Node daemon
- *dynmn_tool disconnect*: to set the Mobile Node to disconnect from waiting for Agent Advertisement messages
- *dynmn_tool tunnel HA*: to register the co-located COA of the MN and set a tunnel directly to the Home Agent.

The command *dynha_tool* followed by command *st 1* can be run on the Home Agent console to view registration and tunnel setup in the Home Agent. The corresponding command *dynmn_tool* followed by *st 1* can also be run on the Mobile Node while the script is running and shows whether the Mobile Node is at its Home Network or Foreign Network. The status information of the Mobile Node shown below resulted after the Mobile Node roamed into the Foreign Network.

Mobile status:

state	Connected
local addr	192.168.0.3
co-addr	192.168.0.3
FA-addr	155.239.170.110
HA-addr	155.239.170.110
Home addr	10.15.185.143
tunnel is	up

lifetime left	296s
tunneling mode	full tunnel direct to HA
last request	4s ago; Wed Oct 20 17:17:24 2004
last reply	3s ago; Wed Oct 20 17:17:25 2004
reply code	0 - registration accepted
info text	connection established
last warning	connected - current_adv == NULL
active	devices 1
discarded msgs	2

The status of the ip routing table of the Mobile Node is updated by the handover script and looks as follows:

Table 5.1 MIP routing table

Kernel IP routing table

Destination	Gateway	Genmask	Flags	Metric	Ref	Use	Iface
155.239.170.110	192.168.0.4	255.255.255.255	UGH	0	0	0	eth1
192.168.0.0	0.0.0.0	255.255.0.0	U	0	0	0	eth1
0.0.0.0 0.	0.0.0 0.0.	0.0.0.0	U	0	0	0	TUNLMNA

The table above shows first two normal routes that could be implemented by any pc connected to the Internet. The 192.168.0.0 network is the Foreign Network, this implies packets sent to a host on the same Foreign Network will be directly routed without tunneling. This is compliant to the specification of RFC 3220. The last route entry shows unfamiliar interface, TUNLMNA. This is a virtual interface created by Mobile IP and it is setup by the script after the detection of the Mobile Node having roamed in the Foreign Network. After the registration of the Mobile Node's co-located COA to the Home Agent, all packets sent to the Correspondent Node with ip address that is not present in the route table will use this route. The function of the TUNLMNA interface is to encapsulate ip packets that consist of Correspondent Node address as destination and Home Address of the Mobile Node as source with an ip header that has the Home Agent address as destination and MN's co-located COA as source. The packets in the testbed that show this mechanism are shown in the following page in *Figure 5.4*.

```
Sniffed at eth1
Root:~# tcpdump –n –i  eth1 ip host 192.168.0.3
08:56:23.340969 155.239.170.110.434 > 192.168.0.3.1248:  udp 72 (DF)
08:56:24.707703 192.168.0.3 > 155.239.170.110: 10.15.190.250.1251 > 196.4.160.2.53:  61613+ A? dev. (21) (DF) (ipip)
08:56:29.710110 192.168.0.3 > 155.239.170.110: 10.15.190.250.1252 > 196.4.160.8.53:  61613+ A? dev. (21) (DF) (ipip)
08:56:34.720099 192.168.0.3 > 155.239.170.110: 10.15.190.250.1251 > 196.4.160.2.53:  61613+ A? dev. (21) (DF) (ipip)
08:56:39.730101 192.168.0.3 > 155.239.170.110: 10.15.190.250.1252 > 196.4.160.8.53:  61613+ A? dev. (21) (DF) (ipip)
08:56:44.746478 192.168.0.3.1248 > 155.239.170.110.434:  udp 78 (DF)
08:56:44.748034 155.239.170.110.434 > 192.168.0.3.1248:  udp 72 (DF)
```

Figure 5.4 IP within IP in MIP

The figure shows the tunnel that has been setup between the Mobile Node and the Home Agent. It can be seen that the packets that are destined for Correspondent Node in the network address 196.4.16.2/24 are first destined to the Home Agent where after the Home Agent will further forward the packets to the Correspondent Node. The last two rows show the registration process between the MN's co-located COA and the Home Agent using UDP port 434.

5.4 Chapter summary

The architecture of the demo setup was shown graphically and an explanation was given on the reasons that lead to its configuration. The main reason for considering this kind of setup is due to the NAT gateway associating private addresses with external routable addresses. An overview of the configuration files for the daemons was given as well as the flowchart of the handover script and its functionalities. The following chapter discusses results of the signal strength handover criteria obtained from the demo setup

6. RESULTS

6.1 Introduction

This chapter discusses results that were obtained from the testbed discussed in the preceding chapter. It presents performance results of MN's initiated handover based on signal strength decision criteria and combinational measurements. In this project combinational measurements are measurements that are added to the signal strength criteria to improve handover delay and rate e.g dwell time and number of counts the signal strength is below the minimum threshold before the decision to make a handover occurs. The performance parameters that will be measured are: handover delay and handover rate. The following are different signal strength criteria that were used in the different measurements:

- P_{new} less than X_{low}. Where P_{new} is the current RSS, and X_{low} is the chosen minimum threshold signal strength. The decision handover script triggers a handover if the current RSS is below the minimum threshold i.e. ($P_{new} < X_{low}$).

- P_{new} less than X_{low} plus T_d, where T_d is the dwell time in seconds. A decision is taken to trigger a handover if the RSS is below the minimum threshold for certain amount of time.

- P_{new} less than X_{low} for N, where N is equal to the number of counts P_{new} remains below X_{low}. The handover script triggers a handover based on the criteria that the RSS is found to be below the minimum threshold for about *N counts*.

6.2 WLAN signal strength

Figure 6.1 shows the received signal strength in dBm experienced by the MN as it moves away from the AP. The MN is capable of communicating efficiently at more than -88dBm but it completely looses signal strength at -92 dBm. It receives a strong signal at approximately -30dbm. *Figure 6.2* also shows that the data link rate is dependent on the signal strength. The figure shows highest and lowest bandwidth achievable in the 802.11b system that was setup. This kind of signal

strength testing mechanism can be implemented to discover higher and lower limits of the RSS for 802.11b Access Points.

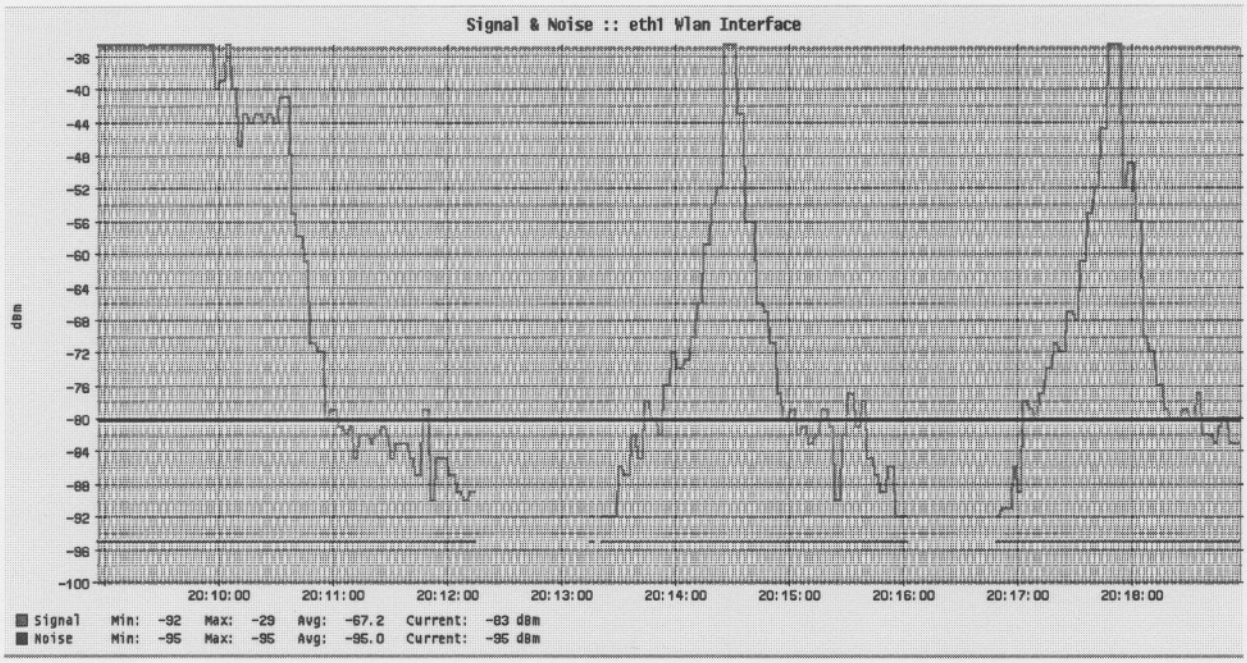

Figure 6.1 Signal Strength and Noise

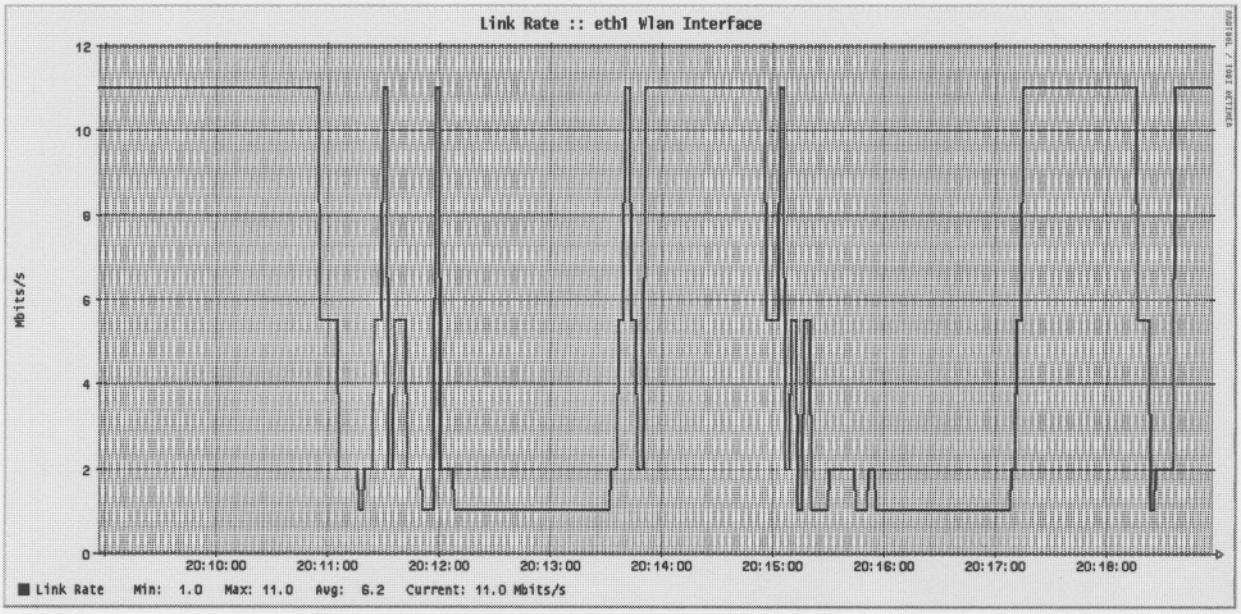

Figure 6.2 WLAN link rate

Figure 6.1 shows that signal strength variation is a very important characteristic of signal strength for the WLAN system experienced by the MN as it moves away from the Access Point. The MN experiences a rapid decrease and increase of the signal strength as it moves out of the WLAN range. This can be seen in *Figure 6.1* just below the thick horizontal black line that crosses point -80dBm on the y-axis. Hence the user experiences inconvenient data rates if changing between HN and FN. It is because of this characteristic that decision timed handovers that are based on signal strength are important, since the user could experience lower and higher data rates and temporary halt of data session in rapid short bursts. Hence signal strength based criteria have to be improved in order to provide a user with convenient handovers as possible.

It was described in the preceding chapters that a tunnel must be established between a MN and HA when using co-located COA Mobile IP architecture to achieve handover. Hence analysis of handovers from one network to the other are done on the start and end of establishment of the tunnel from the MN to the HA. The creation and destruction of the tunnel implies handover between GPRS and 802.11b WLAN networks.

Threshold signal strength of -70dBm is used in the GPRS/802.11b handover test. The -70dBm is selected because it is a safe quantity before the signal could be lost at -92dBm or vary frequently at below -80dBm. Firstly, decision based on solely -70dbm threshold is experimented. Thereafter this decision is implemented in combination with dwell time and number of counts the current signal strength remains below the minimum threshold.

6.3 RSS criteria

The following figures depict graphs of the signal strength and MN's handovers between GPRS and 802.11b WLAN, respectively. *Figure 6.3* shows the RSS experienced by the MN as it moves away and towards the AP. The minimum threshold signal strength for handover is -70 dBm. The tunnel between the HA and the MN has to be destroyed as the MN experiences signal strength below -70 dBm. It has to be established as the MN experiences signal strength above -70 dBm. *Figure 6.4* shows the destruction and establishment of the tunnel as the RSS crosses the -70 dBm line.

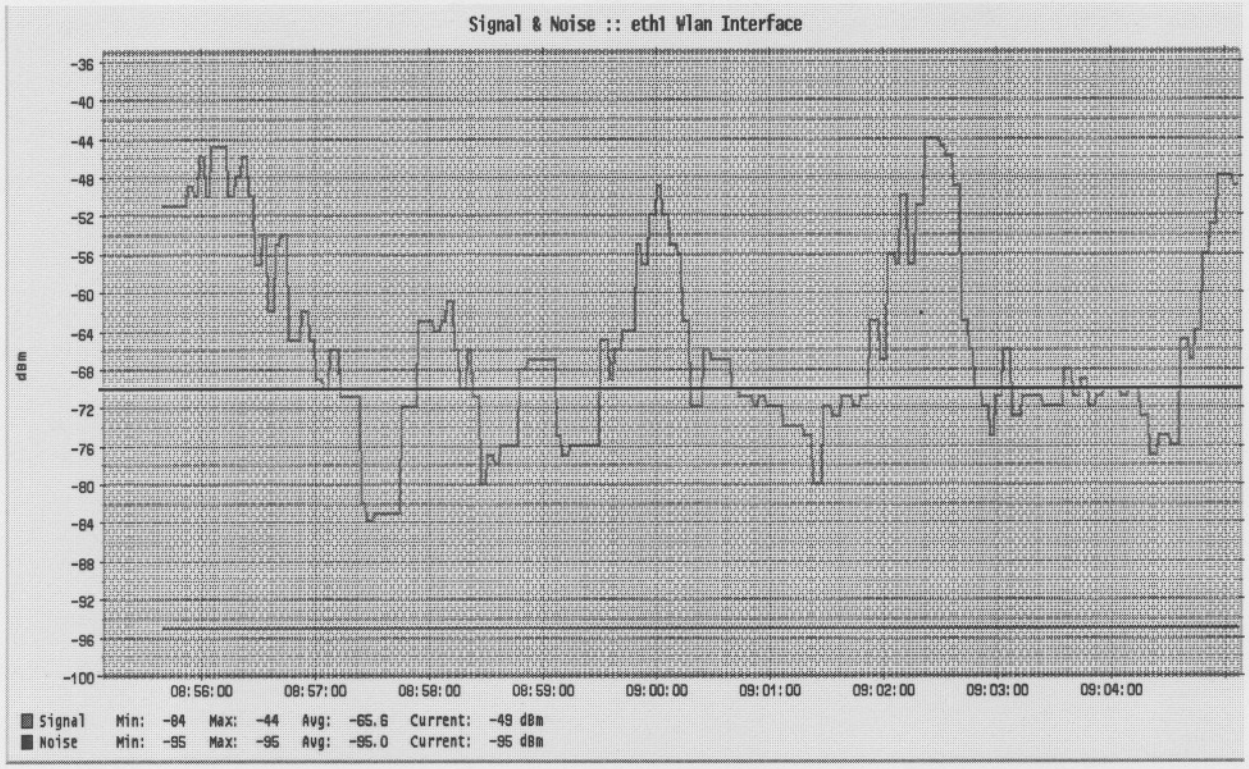

Figure 6.3 P_{new} less than X_{low} RSS

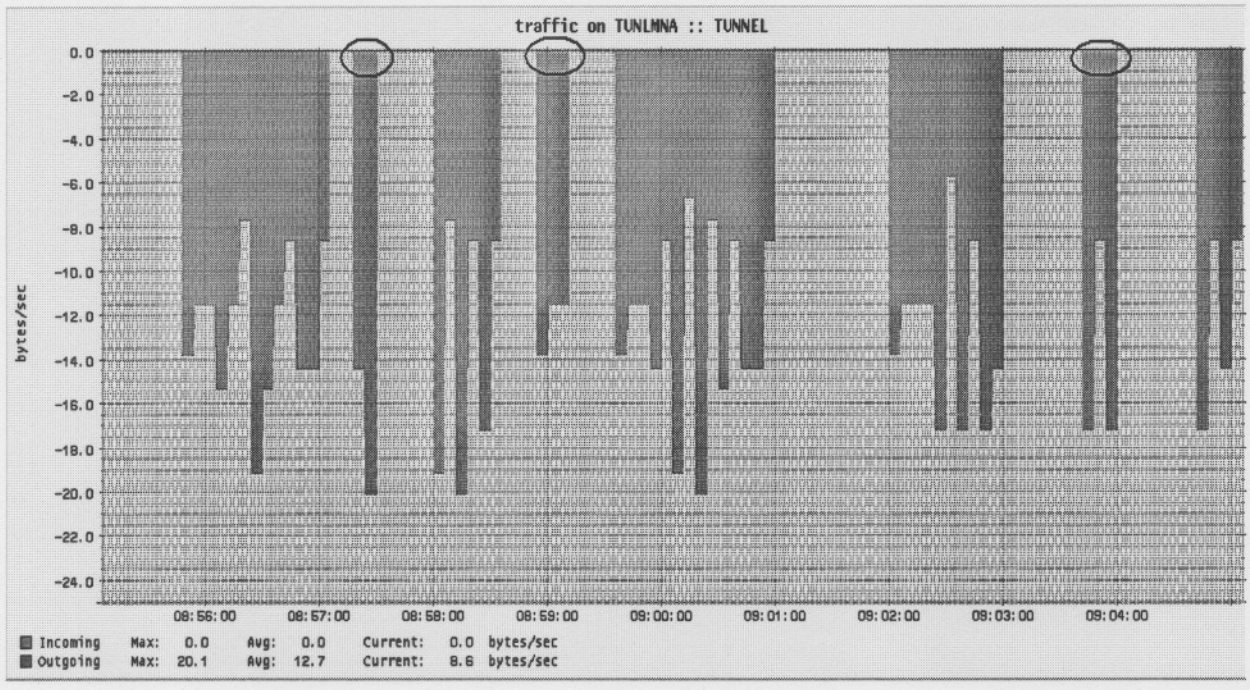

Figure 6.4 P_{new} less than X_{low} handover

Figures 6.3 and *6.4* depict signal strength behaviour and handovers respectively. The handover decision based on $P_{new} < X_{low}$ criteria resulted in a number of short duration handovers as can be seen by circled parts in *Figure 6.4* after 08:57, at 08:59 and before 09:04. This is due to short duration variations in signal strength around the threshold of -70dBm. There were no handover delays that were experienced in changing from GPRS to 802.11b or the other way around. The following discussions on the two remaining criteria focus on eliminating the short duration handovers shown in *Figure 6.4* due to RSS variation around the -70dBm threshold.

6.4 P_{new} less than X_{low} plus T_d criterion

A dwell time was included in the criterion that was only based on signal strength i.e. $P_{new} < X_{low}$ criterion. Different dwell times were selected starting with 1 second up to 3 seconds. The addition of the dwell time dictates that the handover should not be initiated immediately if the signal strength threshold is crossed. Only if it is found that the signal strength is still below the minimum threshold after the dwell time, it is then the handover can be initiated. *Figure 6.5* and *6.6* show results obtained from combining dwell time and criterion based on signal strength.

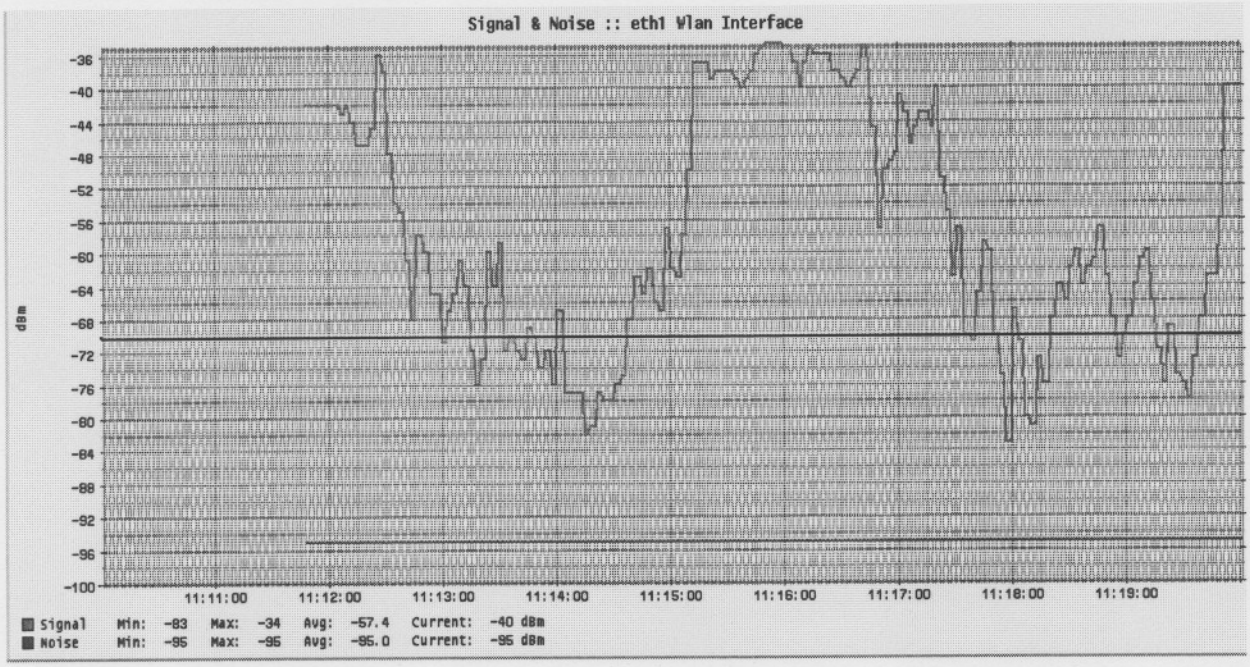

Figure 6.5 P_{new} less than X_{low} plus T_d RSS

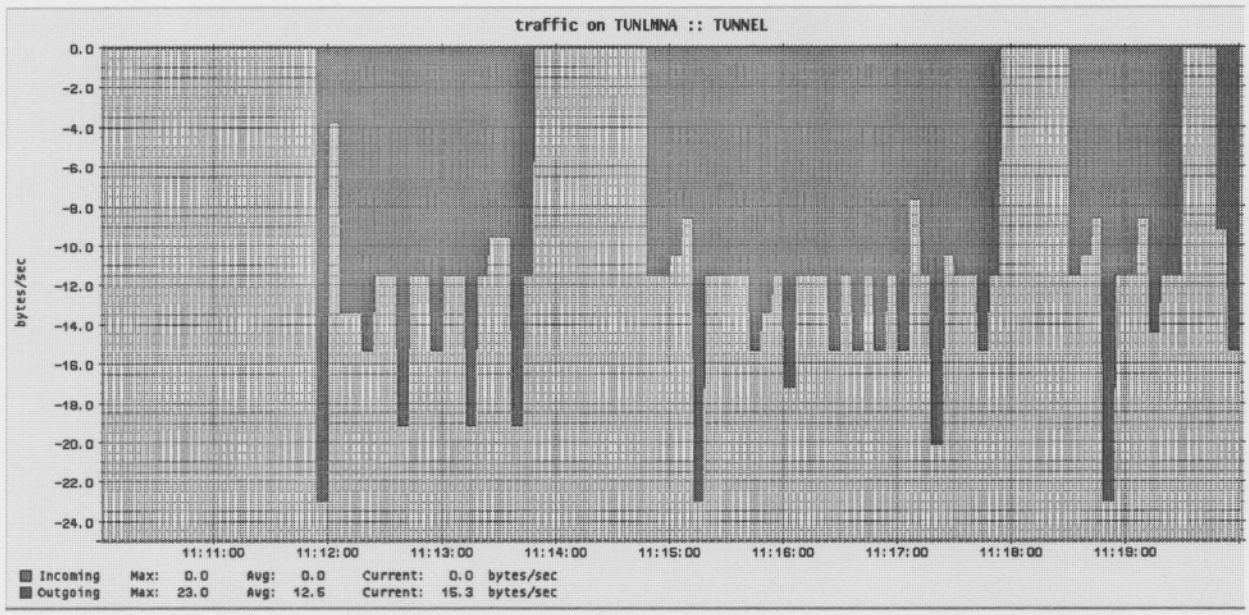

Figure 6.6 P_{new} less than X_{low} plus T_d handover

It can be seen from Figure 6.5 that the signal strength crosses the threshold line about 11 times between points 11:13 and 11:14 and this resulted in one handover after 1 second shown in *Figure 6.6*. This is an improvement in handover as compared to the first criterion. Nevertheless there is a shortcoming to this criterion. The dwell time functioned well for a value of 1 second, but introduced delays as the dwell time was increased to 3 seconds. This method is effective for Mobile Nodes that roam out of range of the WLAN into the GPRS network.

6.5 P_{new} less than X_{low} for N criterion

Figures 6.7 and *6.8* show the signal strength behaviour and handover of the $P_{new} < X_{low}$ *for N counts* handover criterion, N was set to 3. This implies that the handover will be initiated if the threshold signal strength is successively found to be below the threshold in 3 counts. The handover will not occur if it is found that the current RSS is counted to be below the minimum threshold for only once or twice. It can be said that this criterion minimizes handover rate. This is noticed from the two figures by realizing the instant points the RSS crosses the threshold. This is realized between points 13:01 and13:02. There are 11 occurrences in which the RSS is below the threshold and the handover is made only once. This could have resulted in 11 unwanted handovers for criterion solely

based on $P_{new} < X_{low}$ decision. It can also be realized that there are more than 3 crossings of the threshold between points 13:04 and 13:05 which result in no handover. There were no delays in handover as compared to the second criterion of dwell time even when N was increased beyond 3. Hence the unwanted handovers and delays are minimized as compared to the other two preceding experiments.

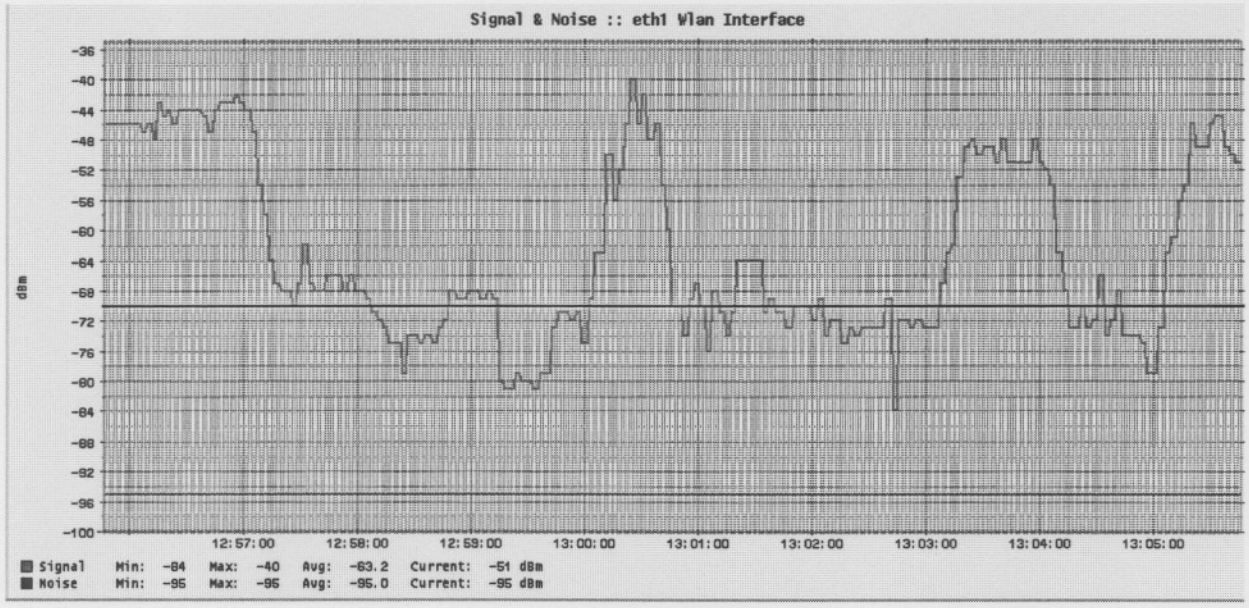

Figure 6.7 P_{new} less than X_{low} for N RSS

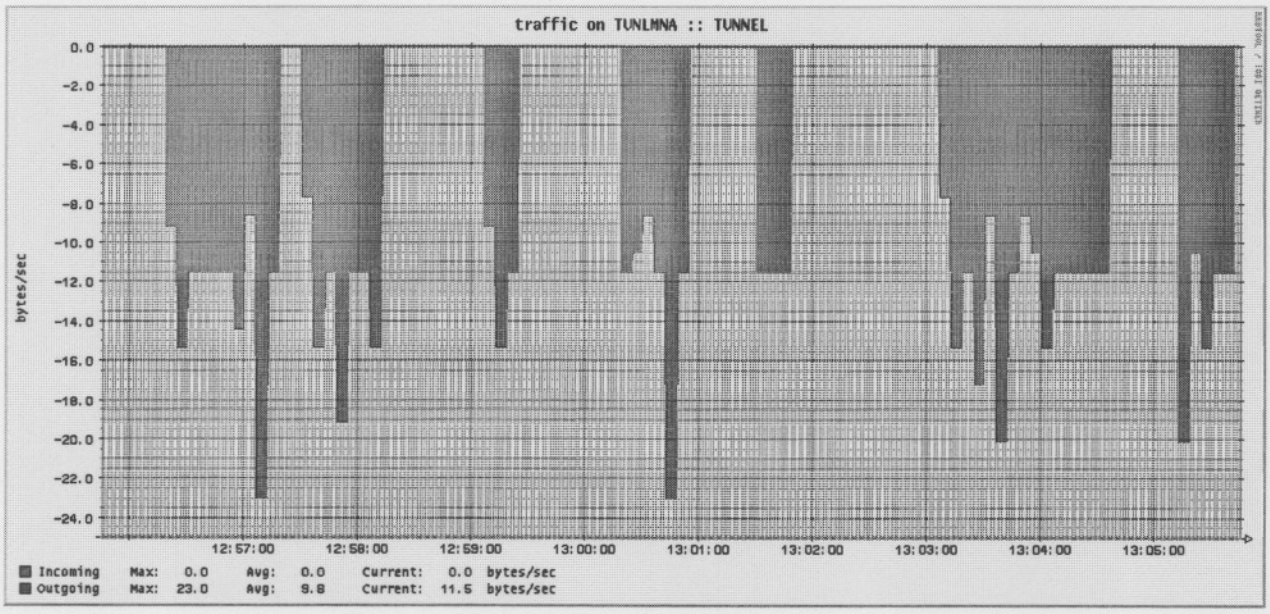

Figure 6.8 P_{new} less than X_{low} for N handover

6.6 Chapter summary

In this chapter the results obtained from the testbed were discussed and parameters of delay and handover rate were shown by different graphs for each criterion. It was also explained that the use of the dwell time with RSS criteria introduces delays in initiating a handover and these delays are proportional to the dwell time. Again the $P_{new} < X_{low}$ *for* N criterion proved to be the one that is capable of reducing handover rate with no delays as compared to the dwell time criterion.

7. CONCLUSION

7.1 Introduction

A need to achieve mobility increases as more advanced mobile devices with capabilities to use different access networks to access Internet become vast. It can be said that in years to come, these mobile devices will have capabilities to switch from one technology to the other while engaged in a data transfer session. Hence continuing research on integration of hybrid networks is very important. It is a reality in the world that wireless LAN technology will affect wireline telecommunication and mobile operators as WLAN data rates increase and access is at affordable cost.

There is also a proliferation of Hot-Spots in many countries including South Africa, which will incite a need in consumers to acquire WLAN enabled devices. Nevertheless WLAN technology has a shortcoming in the communication distance or coverage. Cellular technology on the other hand offers ubiquitous coverage over large areas. The introduction of GPRS has set a milestone in pursuit of cellular technology to offer friendly and affordable Internet access but at low data rates compared to WLAN.

7.2 Purpose of the project

The main goal of the research was to evaluate the performance of GPRS/802.11b Mobile-Node initiated handover based on signal strength criteria. There are three signal strength handover criteria that were implemented in the GPRS/802.11b testbed and performance evaluation was judged on delay and handover rate as discussed in Chapter 6. This project utilized the RSS strength metric in combination with dwell time and number of counts the current RSS remains below the minimum threshold.

7.3 Research approach

In Chapter 1 Section 1.4 various issues and topics (to be addressed during the research) were outlined. It included the following:

- What is GPRS and 802.11b?
- Mobile IP principles
- Handover solutions
- Set up of GPRS and 802.11b handover demonstration
- Parameter measurements

The first question followed by two topics was discussed in Chapters 2, 3 and 4, respectively. A broad description regarding these topics was given in these chapters. It was then after the knowledge acquired from these chapters that a GPRS/802.11b testbed was setup as discussed in Chapter 5. Results of signal strength handover criteria are shown graphically in Chapter 6 and their performance is evaluated in terms of handover delay and handover rate. Chapter 6 revealed the performance as well as the shortcomings of the selected criteria.

The criterion solely based on signal strength results in high unwanted handovers. In order to reduce number of unwanted handovers, a dwell time metric was combined with the signal strength criterion. This worked well but at dwell time values that are less than 3 seconds, delays in handover are significant beyond this value. Nevertheless the limitation in this criterion, it reduced unwanted handovers in the testbed. Another criterion that proved to perform better in terms of insignificant delays and reduced handover rates was discussed in Chapter 6. This criterion could be implemented in a way that suits the user of such GPRS/802.11b setup, by increasing or decreasing parameter *N*.

7.4 Future work

The results and testbed in this thesis opens the possibility for future research. The research possibilities are:

- Combined billing in the GPRS/WLAN infrastructure. One bill for use of GPRS and WLAN network is important, since mobile operators would like a share in the WLAN market.

- The impact of integration of WLAN systems in cellular network is one of the research possibilities. The impact can be of cost for integration, decrease or increase in revenue for mobile operators.

- Use of neural network decision algorithms in GPRS/Wi-Fi to improve handover. This research possibility can further improve handover rates.

7.5 Final conclusion

The use of cellular packet-switched technology as in GPRS and the use of WLAN broadband technology is increasing in South Africa. These technologies are now at a stage where most consumers develop a need for them whether at Hot-Spots or anywhere. Hence a need for achieving mobility between hybrid networks also increases. Therefore handover issues that relate to handover in order to achieve mobility between different networks will continue to be of importance to the consumers and the telecommunication sector. In this project a criterion that best achieves handover with less delay was found and this type of research can serve as a basis for possible future projects in evaluating and developing better handover algorithms between hybrid access networks.

Appendix A: Handover Script

```
#!/bin/sh
# The script is designed to be used on a Mobile Node that only allows a
# co-located care-of address (COA), so it will never try to find and use
# a Foreign Agent.

# This file starts with the the declaration and definition of some
# variables and functions. These will be used in the script
# that follows. The script is a 'never ending' loop. When the script
# is terminated with ^c, the route table will very likely be corrupted.
# A re-arrangement of the routing table will update the table corectly.
#
# Get the fixed home IP-address of the Mobile Node
# and the IP-address of the Home Agent from
# the dynmnd.conf configuration file.
# Therefore the Mobile Node functionality of
# HUT Dynamics Mobile IPv4 0.8.1 has to be implemented on the laptop machine.
MNHomeIPAddress=`cat /usr/local/etc/dynmnd.conf | egrep '^MNHomeIPAddress' |
cut -d " " -f 2`
HAIPAddress=`cat /usr/local/etc/dynmnd.conf | egrep '^HAIPAddress' | cut -d "
" -f 2`
# Definition of several arrays, one for each possible eth or ppp interface.
# Each array has 3 values: IP-address, subnet or ppp peer and default
gateway.
# The latest information will be stored in the "NEW" arrays.
# Reference information is stored in "OLD" arrays.
declare -a eth0NEW
declare -a eth1NEW
declare -a eth2NEW
declare -a ppp0NEW
declare -a ppp1NEW
# All NEW arrays together form the 'NEW state'.
declare -a eth0OLD
declare -a eth1OLD
declare -a eth2OLD
declare -a ppp0OLD
declare -a ppp1OLD
# All OLD arrays together form the 'OLD state'.
# To keep track of a one-time change, a variable TUN_MODE_SET is used:
TUN_MODE_SET=0
# FILL_NEW_ARRAYS scans the current IP configuration and routing table and
# writes specific parameters to a temporary file which will be used to check
# if
# something has changed since previous check.
# The variable IP_ROUTE_SNAPSHOT is used as a buffer to make sure that all
# parameters are captured at the same time, so no changes will be encountered
# during the filling of the members of the NEW arrays:
# 0: IP-address
# 1: subnet/ppp peer
# 2: default gateway
## Sometimes the default gateway will be found a fraction of a second later
## than the IP-address and the subnet.
## To make sure that all changes have been captured, the filling of the NEW
## variables will be delayed by 0.2 seconds, after a subnet change has been
```

```
## detected.
## After those 0.2 seconds a new capture snapshot will be taken, which should
## contain the default gateway now.
## NOTE: It is very important that within these 0.2 seconds no interfaces are
ad#ded
## or removed, because only one change at a time can be Handled by this
function#!!!
## This is done in the first if-loop of FILL_NEW_ARRAYS
function FILL_NEW_ARRAYS {
IP_ROUTE_SNAPSHOT=`ip route`
if [ \
"`echo "$IP_ROUTE_SNAPSHOT" | grep '^[^d].*eth0.*Kernel' | awk '{ print $1
}'`" !=
"${eth0NEW[1]}" -o \
"`echo "$IP_ROUTE_SNAPSHOT" | grep '^[^d].*eth1.*Kernel' | awk '{ print $1
}'`" !=
"${eth1NEW[1]}" -o \
"`echo "$IP_ROUTE_SNAPSHOT" | grep '^[^d].*eth2.*Kernel' | awk '{ print $1
}'`" !=
"${eth2NEW[1]}" ]; then
#echo "sleep 0.2 s"
sleep 0.2s
IP_ROUTE_SNAPSHOT=`ip route`
fi
if [ "`echo "$IP_ROUTE_SNAPSHOT" | grep '^[^d].*eth0.*Kernel' | awk '{ print
$1 }'`" !=
"${eth0NEW[1]}" ]; then
# Only change the default gateway if the subnet has changed,
# otherwise EVAL_GWS will not function correctly.
echo The subnet of eth0 has changed, so write new default gateway.
eth0NEW[2]=`echo "$IP_ROUTE_SNAPSHOT" | grep '^default.*eth0' | cut -d " " -f
3`
fi
eth0NEW[0]=`echo "$IP_ROUTE_SNAPSHOT" | grep '^[^d].*eth0.*Kernel' | awk '{
print $9 }'`
eth0NEW[1]=`echo "$IP_ROUTE_SNAPSHOT" | grep '^[^d].*eth0.*Kernel' | awk '{
print $1 }'`
if [ "`echo "$IP_ROUTE_SNAPSHOT" | grep '^[^d].*eth1.*Kernel' | awk '{ print
$1 }'`" !=
"${eth1NEW[1]}" ]; then
# Only change the default gateway if the subnet has changed,
# otherwise EVAL_GWS will not function correctly.
echo The subnet of eth1 has changed, so write new default gateway.
eth1NEW[2]=`echo "$IP_ROUTE_SNAPSHOT" | grep '^default.*eth1' | cut -d " " -f
3`
fi
eth1NEW[0]=`echo "$IP_ROUTE_SNAPSHOT" | grep '^[^d].*eth1.*Kernel' | awk '{
print $9 }'`
eth1NEW[1]=`echo "$IP_ROUTE_SNAPSHOT" | grep '^[^d].*eth1.*Kernel' | awk '{
print $1 }'`
if [ "`echo "$IP_ROUTE_SNAPSHOT" | grep '^[^d].*eth2.*Kernel' | awk '{ print
$1 }'`" !=
"${eth2NEW[1]}" ]; then
# Only change the default gateway if the subnet has changed,
# otherwise EVAL_GWS will not function correctly.
echo The subnet of eth2 has changed, so write new default gateway.
```

```
eth2NEW[2]=`echo "$IP_ROUTE_SNAPSHOT" | grep '^default.*eth2' | cut -d " " -f
3`
fi
eth2NEW[0]=`echo "$IP_ROUTE_SNAPSHOT" | grep '^[^d].*eth2.*Kernel' | awk '{
print $9 }'`
eth2NEW[1]=`echo "$IP_ROUTE_SNAPSHOT" | grep '^[^d].*eth2.*Kernel' | awk '{
print $1 }'`


ppp0NEW[0]=`echo "$IP_ROUTE_SNAPSHOT" | grep '^[^d].*ppp0.*Kernel' | awk '{
print $9 }'`
ppp0NEW[1]=`echo "$IP_ROUTE_SNAPSHOT" | grep '^[^d].*ppp0.*Kernel' | awk '{
print $1 }'`
ppp0NEW[2]=`echo "$IP_ROUTE_SNAPSHOT" | grep '^default.*ppp0' | cut -d " " -f
3`
ppp1NEW[0]=`echo "$IP_ROUTE_SNAPSHOT" | grep '^[^d].*ppp1.*Kernel' | awk '{
print $9 }'`
ppp1NEW[1]=`echo "$IP_ROUTE_SNAPSHOT" | grep '^[^d].*ppp1.*Kernel' | awk '{
print $1 }'`
ppp1NEW[2]=`echo "$IP_ROUTE_SNAPSHOT" | grep '^default.*ppp1' | cut -d " " -f
3`
}
# EVAL_GWS evaluates the gateways that have been found during FILL_NEW_ARRAYS
# If no default gateway was found, EVAL_GWS guesses the gateway based on 2
princ#iples:
# 1: If an eth device misses a default gateway, it is assumed that this is #
# becau#se
# another eth device is connected to the same subnet and therefore already
#has
# configured the appropriate gateway. This gateway will be copied to the new
# eth interface.
# 2: If an ppp device misses a default gateway, it is assumed that this is
becau#se
# another interface has already set a default gateway. The gateway will be
set b#y
# copying the peer IP-address, wich is the default gateway for that
interface.
function EVAL_GWS {
# Check whether eth0 has a default gateway.
# First check whether eth0 is up and whether a default gateway has already
been #found:
if [ "${eth0NEW[0]}" != "" -a "${eth0NEW[2]}" = "" ]; then
# eth0NEW[0] is not empty, so eth0 has an IP address and thus is up!
# AND
# no default gateway was found
# check for other eth interfaces with same subnet:
if [ "${eth0NEW[1]}" = "${eth1NEW[1]}" ]; then
# etH0 is on the same subnet as eth1 and therefore gets the same default
gateway#:
eth0NEW[2]=${eth1NEW[2]}
elif [ "${eth0NEW[1]}" = "${eth2NEW[1]}" ]; then
# eth0 is on the same subNet as eth2 and therefore gets the same default
gateway#:
eth0NEW[2]=${eth2NEW[2]}
fi
fi
# Do the same for eth1:
```

```
if [ "${eth1NEW[0]}" != "" -a "${eth1NEW[2]}" = "" ]; then
if [ "${eth1NEW[1]}" = "${eth0NEW[1]}" ]; then
eth1NEW[2]=${eth0NEW[2]}
elif [ "${eth1NEW[1]}" = "${eth2NEW[1]}" ]; then
eth1NEW[2]=${eth2NEW[2]}
fi
fi
# Do the same for eth2:
if [ "${eth2NEW[0]}" != "" -a "${eth2NEW[2]}" = "" ]; then
if [ "${eth2NEW[1]}" = "${eth0NEW[1]}" ]; then
eth2NEW[2]=${eth0NEW[2]}
elif [ "${eth2NEW[1]}" = "${eth1NEW[1]}" ]; then
eth2NEW[2]=${eth1NEW[2]}
fi
fi
# Now for the ppp interfaces:
# First check whether ppp0 is up and whether a default gateway has already
been #found:
if [ "${ppp0NEW[0]}" != "" -a "${ppp0NEW[2]}" = "" ]; then
# ppp0 is up AND no gateway was found
# so guess that the defaulte gateway is the ppp peer
ppp0NEW[2]=${ppp0NEW[1]}
fi
# Do the same for ppp1:
if [ "${ppp1NEW[0]}" != "" -a "${ppp1NEW[2]}" = "" ]; then
ppp1NEW[2]=${ppp1NEW[1]}
fi
}
# COPY_ARRAYS copies the NEW arrays to the OLD arrays.
# The OLD arrays can be seen as reference values to keep track of changes.
function COPY_ARRAYS {
eth0OLD[0]=${eth0NEW[0]}
eth0OLD[1]=${eth0NEW[1]}
eth0OLD[2]=${eth0NEW[2]}
eth1OLD[0]=${eth1NEW[0]}
eth1OLD[1]=${eth1NEW[1]}
eth1OLD[2]=${eth1NEW[2]}
eth2OLD[0]=${eth2NEW[0]}
eth2OLD[1]=${eth2NEW[1]}
eth2OLD[2]=${eth2NEW[2]}
ppp0OLD[0]=${ppp0NEW[0]}
ppp0OLD[1]=${ppp0NEW[1]}
ppp0OLD[2]=${ppp0NEW[2]}
ppp1OLD[0]=${ppp1NEW[0]}
ppp1OLD[1]=${ppp1NEW[1]}
ppp1OLD[2]=${ppp1NEW[2]}
}
# COMPARE_ARRAYS compares the NEW arrays with the OLD reference arrays to see
# if some interface has been added, changed or removed.
# The function returns the following value:
# nothing changed
# OR
# something changed
function COMPARE_ARRAYS {
if [ \
"${eth0NEW[0]}" = "${eth0OLD[0]}" -a "${eth0NEW[1]}" = "${eth0OLD[1]}" -a
"${eth0NEW[2]}" =
```

```
"${eth0OLD[2]}" -a \
"${eth1NEW[0]}" = "${eth1OLD[0]}" -a "${eth1NEW[1]}" = "${eth1OLD[1]}" -a
"${eth1NEW[2]}" =
"${eth1OLD[2]}" -a \
"${eth2NEW[0]}" = "${eth2OLD[0]}" -a "${eth2NEW[1]}" = "${eth2OLD[1]}" -a
"${eth2NEW[2]}" =
"${eth2OLD[2]}" -a \
"${ppp0NEW[0]}" = "${ppp0OLD[0]}" -a "${ppp0NEW[1]}" = "${ppp0OLD[1]}" -a
"${ppp0NEW[2]}" =
"${ppp0OLD[2]}" -a \
"${ppp1NEW[0]}" = "${ppp1OLD[0]}" -a "${ppp1NEW[1]}" = "${ppp1OLD[1]}" -a
"${ppp1NEW[2]}" =
"${ppp1OLD[2]}" \
]; then
echo nothing changed
else
echo something changed
fi
}
# HOME_IF checks whether one of the interfaces has the Mobile Nodes Home IP-
addr#ess.
# The function returns the following value:
# none: there is no interface which has the home IP-address
# OR
# <if>: this interface has the home IP-address
function HOME_IF {
if [ "${eth0NEW[0]}" = "$MNHomeIPAddress" ]; then
echo eth0
elif [ "${eth1NEW[0]}" = "$MNHomeIPAddress" ]; then
echo eth1
elif [ "${eth2NEW[0]}" = "$MNHomeIPAddress" ]; then
echo eth2
elif [ "${ppp0NEW[0]}" = "$MNHomeIPAddress" ]; then
echo ppp0
elif [ "${ppp1NEW[0]}" = "$MNHomeIPAddress" ]; then
echo ppp1
else
echo none
fi
}
# PRINT_ARRAYS prints the current status of all NEW and OLD arrays on the
screen
function PRINT_ARRAYS {
echo '--------------------------------------'
echo 'NEW state at '`date`:
echo ' eth0 IP = '${eth0NEW[0]}
echo ' eth0 subnet = '${eth0NEW[1]}
echo ' eth0 gateway = '${eth0NEW[2]}
echo ' eth1 IP = '${eth1NEW[0]}
echo ' eth1 subnet = '${eth1NEW[1]}
echo ' eth1 gateway = '${eth1NEW[2]}
echo ' eth2 IP = '${eth2NEW[0]}
echo ' eth2 subnet = '${eth2NEW[1]}
echo ' eth2 gateway = '${eth2NEW[2]}
echo ' ppp0 IP = '${ppp0NEW[0]}
echo ' ppp0 peer = '${ppp0NEW[1]}
echo ' ppp0 gateway = '${ppp0NEW[2]}
```

```
echo ' ppp1 IP = '${ppp1NEW[0]}
echo ' ppp1 peer = '${ppp1NEW[1]}
echo ' ppp1 gateway = '${ppp1NEW[2]}
echo 'OLD state:'
echo ' eth0 IP = '${eth0OLD[0]}
echo ' eth0 subnet = '${eth0OLD[1]}
echo ' eth0 gateway = '${eth0OLD[2]}
echo ' eth1 IP = '${eth1OLD[0]}
echo ' eth1 subnet = '${eth1OLD[1]}
echo ' eth1 gateway = '${eth1OLD[2]}
echo ' eth2 IP = '${eth2OLD[0]}
echo ' eth2 subnet = '${eth2OLD[1]}
echo ' eth2 gateway = '${eth2OLD[2]}
echo ' ppp0 IP = '${ppp0OLD[0]}
echo ' ppp0 peer = '${ppp0OLD[1]}
echo ' ppp0 gateway = '${ppp0OLD[2]}
echo ' ppp1 IP = '${ppp1OLD[0]}
echo ' ppp1 peer = '${ppp1OLD[1]}
echo ' ppp1 gateway = '${ppp1OLD[2]}
echo '---------------------------------------'
}
# PRIORITY_IF returns the interface which has the highest priority.
# This means that through this interfcae the tunnel to the
# Home Agent will be set up, if the Mobile Node is not at the Home
# Network.
#
function PRIORITY_IF {
# First look for new added eth interfaces:


#NOISE=`iwconfig eth1|grep Signal|cut -d":" -f4|cut -d" " -f1`
SNR=`iwconfig eth1|grep Quality|cut -d":" -f2|cut -d"/" -f1`
#RATE=`iwconfig eth1|grep Rate|cut -d"M" -f1|cut -b20-24`
SIGNAL=`iwconfig eth1|grep Signal|cut -d":" -f3|cut -d" " -f1`

i=0
j=0

if [ "${SIGNAL}" -gt -70 ]; then
route add -host 155.239.170.110 dev eth1
route del default dev ppp0
echo eth1

elif [ "${SIGNAL}" -le -70 ]; then
#i=0
#j=0
        while [ "${i}" != "5" ];do

              SIGNAL=`iwconfig eth1|grep Signal|cut -d":" -f3|cut -d" " -f1`

                  if [ "${SIGNAL}" -le -70 ]; then
                       i=`expr $i + 1`
                       j=`expr $j + 1`

                        sleep 1s
                  else
                        i=5
```

```
                    j=1
                fi
        #done

if [ $i -eq 5 -a $j -ne 5 ] ; then
route add -host 155.239.170.110 dev eth1
route del default dev ppp0
echo eth1

elif [ $i -eq 5 -a $j -eq 5 ] ; then
route del -host 155.239.170.110 dev eth1
route add default ppp0
echo ppp0
fi
done

# There is no interface available:
else
echo none
fi
}
# READ_MAN_SETTINGS reads the manual preferred interface settings
# (i.e. preferred interface mode and the preferred interface)
# from the file MIPscriptPREF_FILE that is written by the script
MIPscriptPREF.
# It writes these values to variables.
function READ_MAN_SETTINGS {
PREFIFMODE_NEW=`cat ./MIPscriptPREF_FILE|egrep "mode"|cut -d " " -f 4`
#echo PREFIFMODE_NEW = $PREFIFMODE_NEW
PREFIF_NEW=`cat ./MIPscriptPREF_FILE|egrep "Manual"|cut -d " " -f 4`
#echo PREFIF_NEW = $PREFIF_NEW
}
# COMPARE_MAN_SETTINGS compares the current manual preferred interface
settings with
# the reference (OLD) manual overrule settings.
function COMPARE_MAN_SETTINGS {
if [ "$PREFIFMODE_NEW" = "$PREFIFMODE_OLD" -a "$PREFIF_NEW" = "$PREFIF_OLD"
]; then
echo "nothing changed"
else
echo "something changed"
fi
}
# COPY_MAN_SETTINGS will copy the NEW manual preferred interface settings to
# the OLD (reference) ones:
function COPY_MAN_SETTINGS {
PREFIFMODE_OLD=$PREFIFMODE_NEW
PREFIF_OLD=$PREFIF_NEW
}
################################
# The real script starts here: #
################################
###### First we have an initialization phase
# It is defined as a function so it can be easily "turned off" by
# commenting it out.
function INITIALIZE {
# The NEW arrays will be filled for the first time
```

```
FILL_NEW_ARRAYS
EVAL_GWS
# Check for an available interface, if none, quit
if [ "`PRIORITY_IF`" = "none" ]; then
echo Connect to the Internet before running MIPscript,
echo MIPscript will be terminated.
exit
fi
# The reference arrays (OLD) have to be the same for a start:
COPY_ARRAYS
# Check the status:
PRINT_ARRAYS
# Set the NEW and OLD manual preferred interface settings to default values.
# This means that the script will start with the automatic mode:
./MIPscriptPREF a eth0
READ_MAN_SETTINGS
COPY_MAN_SETTINGS
# Kill all current Mobile Node daemons, if any:
killall dynmnd 2> /dev/null
# Start new HUT Dynamics 0.8.1 Mobile Node daemon:
dynmnd
# Only the first time the Mobile Node starts using a Foreign Network,
# the tunneling mode must be set to "tunnel HA", so the daemon
# will never try to use Foreign Agents:
if [ "`HOME_IF`" = "none" ]; then
dynmn_tool disconnect
echo @@@@@@@@@@@@@@@ Set tunneling mode to tunnel-direct-to-Home-Agent:
dynmn_tool tunnel HA
TUN_MODE_SET=1
fi
}
# Run the initialization:
INITIALIZE
###### After the initialization the 'never ending' loop will be entered:
while [ 1 ]; do
FILL_NEW_ARRAYS
EVAL_GWS
READ_MAN_SETTINGS
if [ "$PREFIFMODE_NEW" = "a" ]; then
# automatic mode
if [ "`COMPARE_ARRAYS`" = "nothing changed" ]; then
sleep 1s
else
echo "Something changed in automatic mode:"
PRINT_ARRAYS
PRIORITY_IF
echo 'SignalStrength = ' ${SIGNAL}
echo 'SigNoiseRatio = ' ${SNR}

if [ "`HOME_IF`" != "none" -a "`PRIORITY_IF`" = "ppp0" ]; then
dynmn_tool update `HOME_IF`
# The following part is probably unnecessary: START WASTE
#echo Renew default gateway of home interface:
#routef
#if [ "`HOME_IF`" = "eth0" ]; then
#route add default gw ${eth0NEW[2]} dev eth0
#elif [ "`HOME_IF`" = "eth1" ]; then
```

```
#route add default gw ${eth1NEW[2]} dev eth1
#elif [ "`HOME_IF`" = "eth2" ]; then
#route add default gw ${eth2NEW[2]} dev eth2
#elif [ "`HOME_IF`" = "ppp0" ]; then
#route add default gw ${ppp0NEW[2]} dev ppp0
#elif [ "`HOME_IF`" = "ppp1" ]; then
#route add default gw ${ppp1NEW[2]} dev ppp1
#fi
# END WASTE
echo new route:
route -n
else
# Only the first time the Mobile Node starts using a Foreign Network,
# the tunneling mode must be set to "tunnel HA", so the daemon
# will never try to use Foreign Agents:
#if [ "$TUN_MODE_SET" != "1" ]; then
if [ "$TUN_MODE_SET" != "1" -a "`PRIORITY_IF`" != "ppp0" ]; then
echo 'SignalStrength = ' ${SIGNAL}
dynmn_tool disconnect
echo @@@@@@@@@@@@@@@ Set tunneling mode to tunnel-direct-to-Home-Agent:
dynmn_tool tunnel HA
TUN_MODE_SET=1
fi
echo Priority interface = `PRIORITY_IF`
if [ "`PRIORITY_IF`" = "none" ]; then
echo There is no interface available!
elif [ "`PRIORITY_IF`" = "eth0" ]; then
echo flush route...
routef
echo add route...
route add $HAIPAddress gw ${eth0NEW[2]} dev eth0
echo new route:
route -n
echo dyn update:
dynmn_tool update eth0
elif [ "`PRIORITY_IF`" = "eth1" ]; then
echo flush route...
routef
echo add route...
route add $HAIPAddress gw ${eth1NEW[2]} dev eth1
echo new route:
route -n
echo dyn update:
dynmn_tool update eth1
elif [ "`PRIORITY_IF`" = "eth2" ]; then
echo flush route...
routef
echo add route...
route add $HAIPAddress gw ${eth2NEW[2]} dev eth2
echo new route:
route -n
echo dyn update:
dynmn_tool update eth2
elif [ "`PRIORITY_IF`" = "ppp0" ]; then
echo flush route...
routef
echo add route...
```

```
route add $HAIPAddress gw ${ppp0NEW[2]} dev ppp0
echo new route:
route -n
echo dyn update:
dynmn_tool update ppp0
elif [ "`PRIORITY_IF`" = "ppp1" ]; then
echo flush route...
routef
echo add route...
route add $HAIPAddress gw ${ppp1NEW[2]} dev ppp1
echo new route:
route -n
echo dyn update:
dynmn_tool update ppp1
fi
fi
COPY_ARRAYS
fi
else # PREFIFMODE_NEW = m
# manual mode
if [ "`COMPARE_MAN_SETTINGS`" = "nothing changed" ]; then
sleep 1s
else
echo "Something changed in manual mode:"
PRINT_ARRAYS
if [ "$PREFIF_NEW" = "`HOME_IF`" ]; then
dynmn_tool update `HOME_IF`
# The following part is probably unnecessary: START WASTE
#echo Renew default gateway of home interface:
#routef
#if [ "`HOME_IF`" = "eth0" ]; then
#route add default gw ${eth0NEW[2]} dev eth0
#elif [ "`HOME_IF`" = "eth1" ]; then
#route add default gw ${eth1NEW[2]} dev eth1
#elif [ "`HOME_IF`" = "eth2" ]; then
#route add default gw ${eth2NEW[2]} dev eth2
#elif [ "`HOME_IF`" = "ppp0" ]; then
#route add default gw ${ppp0NEW[2]} dev ppp0
#elif [ "`HOME_IF`" = "ppp1" ]; then
#route add default gw ${ppp1NEW[2]} dev ppp1
#fi
# END WASTE
echo new route:
route -n
else
# Only the first time the Mobile Node starts using a Foreign Network,
# the tunneling mode must be set to "tunnel HA", so the daemon
# will never try to use Foreign Agents:
#if [ "$TUN_MODE_SET" != "1" ]; then
if [ "$TUN_MODE_SET" != "1" -a "`PRIORITY_IF`" != "none" ]; then
dynmn_tool disconnect
echo @@@@@@@@@@@@@@@ Set tunneling mode to tunnel-direct-to-Home-Agent:
dynmn_tool tunnel HA
TUN_MODE_SET=1
fi
echo Preferred interface = $PREFIF_NEW
if [ "$PREFIF_NEW" = "eth0" -a "${eth0NEW[0]}" != "" ]; then
```

```
echo flush route...
routef
echo add route...
route add $HAIPAddress gw ${eth0NEW[2]} dev eth0
echo new route:
route -n
echo dyn update:
dynmn_tool update eth0
elif [ "$PREFIF_NEW" = "eth1" -a "${eth1NEW[0]}" != "" ]; then
echo flush route...
routef
echo add route...
route add $HAIPAddress gw ${eth1NEW[2]} dev eth1
echo new route:
route -n
echo dyn update:
dynmn_tool update eth1
elif [ "$PREFIF_NEW" = "eth2" -a "${eth2NEW[0]}" != "" ]; then
echo flush route...
routef
echo add route...
route add $HAIPAddress gw ${eth2NEW[2]} dev eth2
echo new route:
route -n
echo dyn update:
dynmn_tool update eth2
elif [ "$PREFIF_NEW" = "ppp0" -a "${ppp0NEW[0]}" != "" ]; then
echo flush route...
routef
echo add route...
route add $HAIPAddress gw ${ppp0NEW[2]} dev ppp0
echo new route:
route -n
echo dyn update:
dynmn_tool update ppp0
elif [ "$PREFIF_NEW" = "ppp1" -a "${ppp1NEW[0]}" != "" ]; then
echo flush route...
routef
echo add route...
route add $HAIPAddress gw ${ppp1NEW[2]} dev ppp1
echo new route:
route -n
echo dyn update:
dynmn_tool update ppp1
else
echo The requested interface is not available, no changes made.
fi
fi
COPY_MAN_SETTINGS
fi
# In case an interface changed during manual mode, update reference arra#ys:
COPY_ARRAYS
fi
done
# end
```

Appendix B: Mobile Node configuration

```
# $Id: dynmnd.conf,v 1.56 2001/10/20 13:36:07 jm Exp $
# Mobile Node configuration file
#
# Dynamic hierarchial IP tunnel
# Copyright (C) 1998-2001, Dynamics group
#
# This program is free software; you can redistribute it and/or modify
# it under the terms of the GNU General Public License version 2 as
# published by the Free Software Foundation. See README and COPYING for
# more details.
#
##############################################################################
#
# NOTE!
#       This is an example configuration file designed to give
#       perspective to the system configuration AND to provide
#       a basis for a working simple test environment.
#       The values of some of the parameters may not be the
#       same as the daemon's defaults, so don't get confused.
#
#       To get a minimal test working, you will need to check the
#       following items:
#         * MNHomeIPAddress
#         * HAIPAddress
#         * EnableFADecapsulation
#         * HomeNetPrefix (if using FA decapsulation or
#           dynamics HA address resolution)
#         * SPI and SharedSecret
#       The rest of the items should work with their preset values in
#       most cases and they can be used to fine tune the operations
#       after the basic operation have been tested successfully.
#
##############################################################################
#
# The Mobile Nodes's IP address in the Home Network.
# If using AAA (see UseAAA below), Home Address can be set to 0.0.0.0 in
order
# to request a Home Address from the AAA infrastructure. This requires that
# also MN NAI is configured.
MNHomeIPAddress 10.15.185.143

# The Mobile Node's Network Access Identifier (NAI) [RFC2794]
# If configured, this NAI is used in registration requests to identify the
# mobile user for AAA services.
#
# MNNetworkAccessIdentifier "user@example.com"

# UseAAA < TRUE | FALSE >. TRUE enables AAA extensions (key requests using
# material from AAA, HA and Home Address discovery using AAA, etc.). This
# requires that MN NAI and AAA related items below are configured.
# FALSE disables these extensions.
UseAAA FALSE
```

```
# The IP address of Mobile Node's Home Agent. In case of a private HA address
# this is the address of the surrogate HA. If the HA address is unknown, set
# this to 0.0.0.0 and make sure that HomeNetPrefix is correct for dynamic
# HA address resolution or use AAA to discover HA address. If the HA has
# multiple interfaces, this should be the address of the "public" interface,
# i.e., the one toward default gateway (it has to be reachable from the
foreign
# networks).
HAIPAddress 155.239.170.110

# If the HA has more than one interfaces, HAIPAddress should be configured to
# be the one reachable from the Internet (i.e., from the Foreign Networks the
# MN may visit). To allows MN to detect other HA's interfaces, their IP
# addresses may be configured here. MN will use this list in addition to
# HAIPAddress when determining whether an agent advertisement is from its own
# HA (i.e., when MN is at home). Multiple lines containing different
addresses
# may be used to configure more than one alternative HA address.
AlternativeHAIPAddress 192.168.0.4
# AlternativeHAIPAddress 10.2.3.4

# AllowHomeAddrFromForeignNet < TRUE | FALSE >. TRUE allows AAA to assign
# a Home Agent and Home Address from the Foreign Network (assuming they are
# set to 0.0.0.0 above). FALSE means that both the Home Agent and the home
# address must be from the home domain.
AllowHomeAddrFromForeignNet FALSE

# The following configuration options PrivateHAIPAddress,
PrivateHAIdentifier,
# and HANetworkAccessIdentifier are only used with Home Networks that use
# private IP addresses and a surrogate HA. In other cases they should be left
# commented.

# The private IP address of Mobile Node's Home Agent.
# Needed only, if surrogate HA is used.
# PrivateHAIPAddress 192.168.200.200

# The identifier for the private HA in SHA (unique 32-bit number)
# PrivateHAIdentifier 1

# Home Agent Network Access Identifier (NAI)
# If configured, this NAI is used to match the HA agent advertisements when
# a MN is determining whether it is at home or not. This is mainly used with
# private HA address that may not be globally unique.
#
# HANetworkAccessIdentifier "ha@example.com"

# EnableFADecapsulation < TRUE | FALSE >. TRUE enables a mode where
# the FA decapsulates the IP-within-IP encapsulated IP packets.
# FALSE disables this mode and sets the default mode where the
# MN decapsulates the IP-within-IP encapsulated IP packets.
# With FA decapsulation the MN uses its Home Address in the interface even in
# the Foreign Network and with MN decapsulation MN needs to acquire a
# co-located care-of address from the visited network (this needs an external
# program; see man pages for more information).
# The two modes cannot be used simultaneously.
EnableFADecapsulation FALSE
```

```
# Network address of Home Network (CIDR format: a.b.c.d/prefix_length)
# This is used with FA decapsulation and dynamics HA address resolution. If
# commented, the routing entry is not removed nor added. The home net entry
# may optionally be used with MN decapsulation - see MNDecapsRouteHandling
# option below.
#
# Example: 192.168.242.0/24
HomeNetPrefix 10.0.0.0/24

# Home net default gateway
# This entry can be used to force a gateway that the MN uses when it is
# at home. If this is left commented, the MN tries to use the default route
# that was in use when the program was started.
#
HomeNetGateway 155.239.164.208

##############################################################################
# a SPI (Security Parameter Index) must be defined for every MN.
# It is used for indexing the security association at the Home Agent.
SPI 1000
#
# The SharedSecret is provided as a HEX number string. The shared secret can
# also be given as a character string
# (e.g. character string "ABCDE" corresponds to HEX number string
4142434445).
# Note: RFC 2002 specifies that the default key size is 128 bits (i.e.
# 16 bytes or 32 hex 'characters'). Dynamics supports also other key lengths.
# This shared secret is used with the HA. This must be commented out when
using
# AAA infrastructure for key generation. In this case, the AAA related items
# below must be configured.
# SharedSecret < shared secret >
# SharedSecret 016A352B2F235E
SharedSecret "test"
#
# Authentication algorithm
#    1: MD5/prefix+suffix (a.k.a. keyed-MD5) [RFC 2002]
#    4: HMAC-MD5 [RFC 2104]
#    5: SHA-1 [FIPS 180-1]
#    6: HMAC-SHA1 [RFC 2104]
# Note! MD5/prefix+suffix has known weaknesses and use of HMAC-MD5 is
# recommented. MD5/prefix+suffix algorithm is for backwards compatability
with
# older versions that do not support more secure HMAC-MD5.
AuthenticationAlgorithm 4
#
# Replay prevention method:
#   0: none
#   1: time stamps
#   2: nonces
ReplayMethod 1
#
# Mobile Node may have optional security associations with Foreign
# Agents. If the security association exists an additional Mobile Node -
# Foreign Agent Authentication Extension is added to the registration
requests.
```

```
#
# The following list contains the shared secrets indexed by SPI (and
# Foreign Agent IP address). The algorithm field specifies the method
# used for key distribution (see the list above). The format of the share
# secret field is identical to the one used with the MN-HA security
# association list above.
#
FA_SECURITY_BEGIN
# SPI           FA IP           Alg.    Shared Secret
#2001           192.168.0.1     4       0123456789ABCDEF
#2002           192.168.0.2     4       "eslkfj89jr3hduh3R!as"
FA_SECURITY_END


# MN-AAA Authentication and Challenge/Response [RFC3012]

# If the MN does not have a security association with an FA, it may use AAA
# infrastructure for authentication. If this is used, also MN NAI
# ('MNNetworkAccessIdentifier' above) should be configured.

# SPI to be used in MN-AAA authentication.
# Reserved SPI values:
#   2 = CHAP_SPI, CHAP style authentication using MD5 [RFC 3012]
#   3 = MD5/prefix+suffix [draft-ietf-mobileip-aaa-key-03.txt]
#   4 = HMAC MD5 [draft-ietf-mobileip-aaa-key-03.txt]
# MN-AAA-SPI 12345

# Shared secret for MN-AAA authentication (see 'SharedSecret' above for
format
# instructions)
# MN-AAA-SharedSecret "test"

# Algorithms to be used for MN-AAA authentication and key generation
#   1 = MD5/prefix+suffix (RFC 2002)
#   2 = RADIUS authentication (Sec. 8 of RFC 3012)
#   3 = MD5/prefix+suffix (RFC 2002)  (alias for 1 above)
#   4 = HMAC-MD5 (Sec. 6 of RFC 3012; RFC 2104)
#   5 = SHA-1 (FIPS 180-1)
#   6 = HMAC-SHA1 (RFC 2104)
# Note: with algorithm 2, 'MN-AAA-SPI' should be set to reserved number
# CHAP_SPI (default: 2).
# MN-AAA-AuthenticationAlgorithm 4
# MN-AAA-KeyGenerationAlgorithm 4


##############################################################################
# TunnelingMode < 1 | 2 | 3 | 4 >
# The packets between the MN and a Correspondent Node (CN) can be routed
using
# different routes. This option can be used to select, which mode will be
# selected.
# Possible values:
# 1 = automatic, prefer reverse tunnel (i.e. bi-directional tunnel)
# 2 = automatic, prefer triangle tunnel (i.e. tunnel only in CN->MN
direction)
# 3 = accept only reverse tunnel
# 4 = accept only triangle tunnel
```

```
TunnelingMode 1

# When MN can get its own co-located care-of address and use reverse
tunneling,
# the normal method is to set the default route to the tunnel. This means
that
# all the packets destined to other networks than the current subnet in the
# visited network are send via the HA. If the co-located COA is public, it
can
# be used for sessions that do not need constant IP address (e.g. most of the
# web browsing). The following configuration option specifies the routing
# operation that is used with the co-located COA.
# Possible values:
#   0 = set default route to the tunnel
#   1 = set only the home net route to the tunnel (the above HomeNetPrefix
#       options must be set)
#   2 = do not change the routing entries (i.e. some external means must be
#       used to direct traffic to the tunnel, e.g. manually adding host route
#       to a specific host)
MNDecapsRouteHandling 0

# DefaultTunnelLifetime is the lifetime suggested in registration
# The lifetime is defined in seconds, default value is 300.
# The request timer will be set according to this value. If the FA's agent
# advertisment has a smaller time, it is used instead.
# Special case: 65535 (or more) seconds means unlimited time (the binding
will
# not expire)
# MNDefaultTunnelLifetime [ seconds ]
MNDefaultTunnelLifetime 300

# UDP port to be used for sending registration requests
# Port 434 is allocated for Mobile IP signaling and this should not be
changed
# unless the network is known to use some other port (i.e. all the FAs and
HAs
# must have the same port configured).
UDPPort 434

# Socket priority for signaling sockets (UDP) can be set with SO_PRIORITY to
# allow easier QoS configuration. If this argument is set, the given value is
# used as a priority for the signaling socket. E.g. CBQ class can be used to
# make sure that signaling is not disturbed by other traffic on a congested
# link.
# This feature is still undocumented and can be left commented.
#
# SocketPriority 1

# The log messages are written through syslog service. The facility to be
# used defaults to LOG_LOCAL0, but it can be set with this parameter
# to any of the possible facilities (LOG_AUTHPRIV, LOG_DAEMON, and so on).
# The processing of log messages is defined in /etc/syslog.conf file.
SyslogFacility LOG_DAEMON

# Ignore these interfaces. No agent advertisements are received nor
# agent solicitations sent for these interfaces.
IGNORE_INTERFACES_BEGIN
```

```
lo
dummy0
tunl0
gre0
IGNORE_INTERFACES_END

# Other programs may set routing entries so that the data connection may
# fail. The MN can try to enforce the routes that it believes should be used.
# This operation should currently be used only with FA decapsulation. If the
# route enforcement is activated the MN daemon prevents certain route
changes.
EnforceRoutes FALSE

# MN can be instructed to poll for current AP address when using a wireless
# LAN driver that supports wireless extensions. This can be used to speed up
# handovers when using managed mode (BSS).
# Polling interval is configured in micro seconds
# (i.e., 1000000 equals to 1 second)
# -1 = AP polling disabled
APPollingInterval -1

# MN can be instructed to send periodic agent solicitations to find new FAs.
# Normally, MN uses agent solicitations when it does not have a valid agent
# advertisement. Periodic solicitation occurs even if the connection seems to
# be up. This will cause more broadcast messages and is thus disabled in the
# default configuration, but it can speed up handovers in some environments.
# Solicitation interval is configured in micro seconds (usec)
# (i.e., 1000000 usec equals to 1 second). A rnadom time between 0 and 0.5
# second will be added to solicitation intervals to prevent unwanted
# synchronization of broadcast messages. In addition, solicitations will not
be
# send more often than once per second, so this interval should not be
# configured to be less than 1000000 usec.
# -1 = Periodic agent solicitation disabled
SolicitationInterval -1

##############################################################################
# Mobile Nodes use unix domain sockets to communicate through their API
# interfaces.
# The group and owner must be names as strings, no groupIDs or userIDs are
# allowed. The file permissions are set in octal values like in chmod(1).
# The configuration parameters of the two API sockets are as follows:
MNAPIReadSocketPath "/var/run/dynamics_mn_read"
MNAPIReadSocketGroup "root"
MNAPIReadSocketOwner "root"
MNAPIReadSocketPermissions 0666
#
MNAPIAdminSocketPath "/var/run/dynamics_mn_admin"
MNAPIAdminSocketGroup "root"
MNAPIAdminSocketOwner "root"
MNAPIAdminSocketPermissions 0700
#
# Every configuration file must end to the keyword 'END'.
END
```

Appendix C: Home Agent configuration

```
# $Id: dynhad.conf,v 1.39 2001/10/20 13:36:07 jm Exp $
# Home Agent configuration file
#
# Dynamic hierarchial IP tunnel
# Copyright (C) 1998-2001, Dynamics group
#
# This program is free software; you can redistribute it and/or modify
# it under the terms of the GNU General Public License version 2 as
# published by the Free Software Foundation. See README and COPYING for
# more details.
#
##############################################################################
#
# NOTE!
#     This is an example configuration file designed to give
#     perspective to the system configuration AND to provide
#     a basis for a working simple test environment.
#     The values of some of the parameters may not be the
#     same as the daemon's defaults, so don't get confused.
#
#
##############################################################################
#

# Interfaces to be used for Mobile IP services. Note that you have to
configure
# each interface that may receive or send registration messages.
# interface: name of the interface, e.g. eth0
# ha_disc:
#    0 = do not allow dynamic HA discovery
#    1 = allow dynamic HA discovery with broadcast messages
# agentadv:
#    0 = do not send agent advertisements without agent solicitation
#    1 = send agent advertisements regularly
#   -1 = do not send any (even solicited) agent advertisements
# interval: number of seconds to wait between two agentadvs
#           (if allowed for this interface)
# force_IP_addr: local address to be forced for this interface
#            (can be used to select one of the multiple virtual
#            addresses); if not entered, the primary address of the
#            interface is used
INTERFACES_BEGIN
# interface  ha_disc  agentadv  interval  force_IP_addr
ppp0         1        1         10        155.239.170.110
eth0         1        1         10        192.0.0.4
#eth1        1        1         20        192.168.240.2
INTERFACES_END

# Network Access Identifier (NAI) of this HA
# Unique identifier for this HA. A macro [interface] can be used to get
# the hardware address of an interface in dot-separated format.
# This is needed, if private address space is used in the Home Network.
# NetworkAccessIdentifier "[eth0]@example.com"
```

```
# Surrogate HA IP Address
# This is only needed, if private address space and a surrogate HA are used
in
# the Home Network.
# SHAIPAddress 10.10.10.10

# Private HA Identifier at SHA
# Unique identifier (32-bit number) at SHA for this private HA.
# This is only needed, if private address space and a surrogate HA are used
in
# the Home Network.
# PrivateHAIdentifier 1

# UDP port to listen for registration requests
# The default is 434
UDPPort 434

# Socket priority for signaling sockets (UDP) can be set with SO_PRIORITY to
# allow easier QoS configuration. If this argument is set, the given value is
# used as a priority for the signaling socket. E.g. CBQ class can be used to
# make sure that signaling is not disturbed by other traffic on a congested
# link.
# This feature is still undocumented and can be left commented.
#
# SocketPriority 1

# MaxBindings can be used to restrict the maximum number of Mobile Nodes
# that are concurrently attached to this Home Agent.
# The default is 20.
MaxBindings 20

# The default tunnel lifetime is suggested also by the HA.
# The default lifetime is 500.
HADefaultTunnelLifetime 600

# The Registration error reply interval should be restricted to
# avoid system overloading situations when receiving too much
# incorrect Registration Reply messages.
# The default value for RegErrorReplyInterval is 1 second.
RegErrorReplyInterval 1

# Triangle tunnel means that the packages to MNs are send via the HA, but
# packages from MN are routed directly (i.e. FA use normal IP routing).
# EnableTriangleTunneling < TRUE | FALSE >
EnableTriangleTunneling FALSE

# Reverse tunnel means bi-directional tunneling in which both the packages
# from and to MN are send via HA
# EnableReverseTunneling < TRUE | FALSE >
EnableReverseTunneling TRUE

##############################################################################
# The Home Agent needs to know what kind of security parameters each
# authorized Mobile Node uses. that is why there is a tbale that maps
# (in many-to-many relationship) SPI numbers, or SPI-number ranges to
# IP adresses - or IP-address ranges defined by network adresses and
```

```
# netmasks. The netmask may be defined in two ways: either in
# "bit offset notation" (the third row in the example) or in the
# "dotted decimal notation" (the fifth row in the example below).
# The list of Mobile Node information is separated between two
# keywords: AUTHORIZEDLIST_BEGIN and AUTHORIZEDLIST_END.
#
# < SPI | SPI-range       IP | network/netmask  >
# Example:

AUTHORIZEDLIST_BEGIN
# SPI           IP
#1000            192.168.240.2
#1001            192.168.240.3
#1002            0.0.0.0/0
#11000-11999     192.168.241.4
#12000           192.168.250.0/255.255.255.0
#13000-14000     192.168.251.0/28
1000        10.15.185.143
AUTHORIZEDLIST_END

# The Home Agents needs a security association for each authorized Mobile
# Node. The association includes following information.
#
# SPI (Security Parameter Index): a key for the other fields.
#
# Authentication Algorithm:
#    1: MD5/prefix+suffix (a.k.a. keyed-MD5) [RFC 2002]
#    4: HMAC-MD5 [RFC 2104]
#    5: SHA-1 [FIPS 180-1]
#    6: HMAC-SHA1 [RFC 2104]
# Note! MD5/prefix+suffix has known weaknesses and use of HMAC-MD5 is
# recommented. MD5/prefix+suffix algorithm is for backwards compatability
with
# older versions that do not support more secure HMAC-MD5.
#
# Replay Protection Method:
#    0: none
#    1: timestamps
#    2: nonces
#
# Timestamp tolerance indicates how many seconds the MN's timestamp can
differ
# from the HA's clock. 7 seconds is the recommended default value. This
# tolerance is checked only when timestamps are used for replay protection.
#
# The maximum lifetime for the binding is given in seconds.
# Special case: 65535 (or more) seconds means unlimited time (the binding
will
# not expire)
#
# Shared Secret: a secret data known by MN and HA. It can be given as
# a HEX code string, i.e. two characters (0-F) correspond to one octet.
# The shared secret can also be given as a character string (e.g.
# "ABCDE" corresponds to 4142434445).
# Note: RFC 2002 specifies that the default key size is 128 bits (i.e.
# 16 bytes or 32 hex 'characters'). Dynamics supports also other key lengths.
#
```

```
# The SPI is the key identificator for the rest of the security parameters
# on the same line. SPI number ranges may be assigned the same security
# parameters.
#
# The list of Mobile Node information is separated between two
# keywords: SECURITY_BEGIN and SECURITY_END.
#
SECURITY_BEGIN
#       auth.    replay  timestamp       max             shared
# SPI   alg.     meth.   tolerance       lifetime        secret
1000  4     1      120         600         "test"
#1002    4       2       60              120             01020304050607
#10000   4       1       60              300             016A352B2F235E
#10001   4       1       120             180             0EF42BD234ECCAA2
SECURITY_END
#
##############################################################################
# Home Agent may have optional security associations with Foreign
# Agents. If the security association exists the session key can be
# encrypted with the help of shared secret and thus man-in-the-middle
# style attacks can be prevented. If no security association is set
# for a certain Foreign Agent - Home Agent pair, public key encryption
# (RSA) is used.
#
# When private address space is used, this list must have a security
# association with the surrogate HA instead of the FAs. Possible security
# associations with the FAs are then configured to the SHA.
#
# The following list contains the shared secrets indexed by SPI (and
# Foreign Agent IP address). The algorithm field specifies the method
# used for authentication and key distribution:
#    1: MD5/prefix+suffix (a.k.a. keyed-MD5) [RFC 2002]
#    4: HMAC-MD5 [RFC 2104]
#    5: SHA-1 [FIPS 180-1]
#    6: HMAC-SHA1 [RFC 2104]
# The format of the share secret field is identical to the one used with the
# MN-HA security association list above.
#
FA_SECURITY_BEGIN
# SPI       FA IP       Alg.  Shared Secret
#2001       192.168.0.1 4     0123456789ABCDEF
#2002       192.168.0.2 4     "eslkfj89jr3hduh3R!as"
FA_SECURITY_END
#
# The Highest FA public key can be protected from man-in-the-middle style
# attacks between the HFA and the HA with hash code. The use of this hash
# is optional, but recommented. The HA can have different ways of checking
# the hash code.
# Methods:
#    0: skip the hash code completely (not recommented)
#    1: if the hash code is received, check the public key with it
#    2: require the correct hash code for every registration message
#       with a public key (this may prevent the use of some organizations
#       which do not advertise the hash code)
PublicKeyHashMethod 1
#
##############################################################################
```

```
# The log messages are written through syslog service. The facility to be
# used defaults to LOG_LOCAL0, but it can be set with this parameter
# to any of the possible facilities (LOG_AUTHPRIV, LOG_DAEMON, and so on).
# The processing of log messages is defined in /etc/syslog.conf file.
SyslogFacility LOG_DAEMON

# Home Agents (and Foreign Agents) use unix domain sockets
# to communicate through their API interfaces.
# The group and owner must be names as strings, no groupIDs or userIDs are
# allowed. The file permissions are set in octal values like in chmod(1).
# The configuration parameters of the two API sockets are as follows:
HAAPIReadSocketPath "/var/run/dynamics_ha_read"
HAAPIReadSocketGroup "root"
HAAPIReadSocketOwner "root"
HAAPIReadSocketPermissions 0766
#
HAAPIAdminSocketPath "/var/run/dynamics_ha_admin"
HAAPIAdminSocketGroup "root"
HAAPIAdminSocketOwner "root"
HAAPIAdminSocketPermissions 0700
#
# Every configuration file must end to the keyword 'END'.
END
```

Appendix D: GPRS ppp configuration

(Option's Globetrotter GPRS PC CARD)

```
# This optionfile was generated by pppconfig 2.0.10.
#
#
hide-password
noauth
connect "/usr/sbin/chat -v -f /etc/chatscripts/vodacomGPRS"
debug
/dev/ttyS3
115200
defaultroute
noipdefault
user xxxxx
remotename vodacomGPRS
ipparam vodacomGPRS
nobsdcomp


lcp-echo-interval 0



# This chatfile was generated by pppconfig 2.0.10.
# Please do not delete any of the comments.  Pppconfig needs them.
#
# ispauth PAP
# abortstring
ABORT BUSY ABORT 'NO CARRIER' ABORT VOICE ABORT 'NO DIALTONE' ABORT 'NO DIAL
TONE' ABORT 'NO ANSWER' ABORT DELAYED
# modeminit
'' ATZ

#Added by Sandile
OK-AT-OK AT+CGDCONT=1,"IP","Internet","",1,1

# ispnumber
OK-AT-OK ATDT*99***1#
# ispconnect
CONNECT \d\c
# prelogin

# ispname
# isppassword
# postlogin

# end of pppconfig stuff
```

Appendix E: WLAN configuration

/etc/pcmcia/network.opts

```
# Network adapter configuration
#
# The address format is "scheme,socket,instance,hwaddr".
#
# Note: the "network address" here is NOT the same as the IP address.
# See the Networking HOWTO.  In short, the network address is the IP
# address masked by the netmask.
#
case "$ADDRESS" in
*,*,*,*)
    INFO="Sample private network setup"
    # Transceiver selection, for some cards -- see 'man ifport'
    IF_PORT=""
    # Use BOOTP (via /sbin/bootpc, or /sbin/pump)? [y/n]
    BOOTP="n"
    # Use DHCP (via /sbin/dhcpcd, /sbin/dhclient, or /sbin/pump)? [y/n]
    DHCP="n"
    # If you need to explicitly specify a hostname for DHCP requests
    DHCP_HOSTNAME=""
    # Use PPP over Ethernet (via the pppoe package)? [y/n]
    PPPOE="n"
    # Use WHEREAMI (via the whereami package)? [y/n]
    WHEREAMI="n"
    # Host's IP address, netmask, network address, broadcast address
    IPADDR="192.168.0.3"
    #IPADDR="143.160.11.164"
    NETMASK="255.255.255.0"
    NETWORK="192.168.0.0"
    #BROADCAST="143.160.11.255"
    # Gateway address for static routing
    #GATEWAY="143.160.8.200"
    # Things to add to /etc/resolv.conf for this interface
    DOMAIN=""
    SEARCH=""
    # The nameserver IP addresses specified here complement the
    # nameservers already defined in /etc/resolv.conf.  These nameservers
    # will be added to /etc/resolv.conf automatically when the PCMCIA
    # network connection is established and removed from this file when
    # the connection is broken.
    #DNS_1="143.160.32.1"
    #DNS_2="143.160.8.205"
    #DNS_3=""
    # NFS mounts, should be listed in /etc/fstab
    MOUNTS=""
    # If you need to override the interface's MTU...
    MTU=""
    # For IPX interfaces, the frame type and network number
    IPX_FRAME=""
    IPX_NETNUM=""
    # Run ipmasq? [y/n]  (see the Debian ipmasq package)
```

```
    IPMASQ="n"
    # Extra stuff to do after setting up the interface
    start_fn () { return; }
    # Extra stuff to do before shutting down the interface
    stop_fn () { return; }
    # Card eject policy options
    NO_CHECK=n
    NO_FUSER=n
    ;;
esac

# This tries to use Debian's network setup in /etc/network/interfaces
# if no settings are given higher up in this file.  You can delete it
# if that isn't desired.

is_true $PUMP || is_true $BOOTP || is_true $DHCP || is_true $DHCLIENT || \
if [ ! "$IPADDR" -a -f /etc/network/interfaces ] ; then
    INFO="Debian network setup"
    start_fn () {
        log /sbin/ifup $1
    }
    stop_fn () {
        log /sbin/ifdown $1
    }
fi
```

/etc/pcmcia/wireless.opts

```
# Wireless LAN adapter configuration
#
# Theory of operation :
#
# The script attempts to match a block of settings to the specific wireless
# card inserted, the *first* block matching the card is used.
 # The address format is "scheme,socket,instance,hwaddr", with * as a
wildcard.
# 'scheme' is the pcmcia scheme (set via 'cardctl scheme XXX').
# 'hwaddr' is the unique MAC address identifier of the wireless card.
# The MAC address is usually printed on the card, or can be found via
ifconfig.
# Some examples here use only half of the MAC address with a wildcard to
# match a whole family of cards...
#
# All the Wireless specific configuration is done through the Wireless
# Extensions, so we will just call 'iwconfig' with the right parameters
# defined below.
# Of course, you need to have iwconfig installled on your system.
# To download iwconfig, or for more info on Wireless Extensions :
#       http://www.hpl.hp.com/personal/Jean_Tourrilhes/Linux/Tools.html
#
# Note : you don't need to fill all parameters, leave them blank, in most
# cases the driver will initialise itself with sane defaults values or
# automatically figure out the value... And no drivers do support all
# possible settings...
#
# If you make any mistakes, you'll get a cryptic message in the system
# log.  You'll need to figure out on your own which parameter was wrong:
#     cardmgr[310]: executing: './network start wvlan0'
#     cardmgr[310]: + SIOCSIWMODE: Invalid argument
# I've tried to give more troubleshooting help at :
#       http://www.hpl.hp.com/personal/Jean_Tourrilhes/Linux/Tools.html#debug
# In case of doubts, just check "/etc/pcmcia/wireless" for the gory
details...
#
# Note also that this script will work only with the original Pcmcia scripts,
# and not with the default Red Hat scripts. Send a bug report to Red Hat ;-)
#
# Finally, send comments and flames to me, Jean Tourrilhes <jt@hpl.hp.com>
#

case "$ADDRESS" in

# NOTE : Remove the following four lines to activate the samples below ...
# --------- START SECTION TO REMOVE -----------
*,*,*,*)
    ;;
# ---------- END SECTION TO REMOVE ------------

# Here is an example of scheme matching
# Activate with "cardctl scheme essidany"

# Pick up any Access Point, should work on most 802.11 cards
```

```
essidany,*,*,*)
    INFO="Any ESSID"
    ESSID="any"
    ;;

# Here are a few examples with a few Wireless LANs supported...
# The matching is done on the first 3 bytes of the MAC address

# Lucent Wavelan IEEE (+ Orinoco, RoamAbout and ELSA)
# Note : wvlan_cs driver only, and version 1.0.4+ for encryption support
*,*,*,00:60:1D:*|*,*,*,00:02:2D:*)
    INFO="Wavelan IEEE example (Lucent default settings)"
    ESSID="AP11B"
    MODE="Managed"
#    RATE="auto"
    KEY="s:secu1"
# To set all four keys, use :
#   KEY="s:secu1 [1] key s:secu2 [2] key s:secu3 [3] key s:secu4 [4] key [1]"
# For the RG 1000 Residential Gateway: The ESSID is the identifier on
# the unit, and the default key is the last 5 digits of the same.
#   ESSID="084d70"
#   KEY="s:84d70"
    ;;

# Cisco/Aironet 4800/340
# Note : MPL driver only (airo/airo_cs), version 1.3 or later
*,*,*,00:40:96:*)
    INFO="Cisco/Aironet example (Cisco default settings)"
    ESSID="any"
# To set all four ESSID, use iwconfig v21 and the same trick as above
    MODE="Managed"
#    RATE="11M auto"
#    KEY="off"
    ;;

# Samsung MagicLan (+ some other PrismII cards)
# Note : Samsung binary library driver, version 1.20 or later
*,*,*,00:00:F0:*|*,*,*,00:02:78:*)
    INFO="Samsung MagicLan example (Samsung default settings)"
    ESSID="any"
    MODE="Managed"
    CHANNEL="4"
    RATE="auto"
#    KEY="883e-aa67-21 [1] key 5501-d0da-87 [2] key 91f5-3368-6b [3] key
2d73-31b7-96 [4]"
#    IWCONFIG="power on"
    ;;

# Raytheon Raylink/WebGear Aviator2.4
# Note : doesn't work yet, please use for debugging only :-(
*,*,*,00:00:8F:*|*,*,*,00:00:F1:*)
    INFO="Raylink/Aviator2.4 example (Aviator default ad-hoc setting)"
    ESSID="ADHOC_ESSID"
    MODE="Ad-Hoc"
    RATE="auto"
    IWPRIV="set_framing 1"
    ;;
```

```
# Old Lucent Wavelan
*,*,*,08:00:0E:*)
    INFO="Wavelan example (Lucent default settings)"
    NWID="0100"
    MODE="Ad-Hoc"
    FREQ="2.425G"
    KEY="off"
    ;;

# Netwave (Xircom Netwave/Netwave Airsurfer)
*,*,*,00:80:C7:*)
    INFO="Netwave example (Netwave default settings)"
    NWID="100"
    KEY="00"
    ;;

# Proxim RangeLan2/Symphony (what is the MAC address ???)
*,*,*,XX:XX:XX:*)
    INFO="Proxim RangeLan2/Symphony example"
    NWID="0"
    MODE="Master"
    CHANNEL="15"
    IWPRIV="setsubchan 1"
    ;;

# No Wires Needed Swallow 550 and 1100 setting (what is the MAC address ???)
*,*,*,XX:XX:XX:*)
    INFO="NWN Swallow example"
    ESSID="session"
    KEY="0000-0000-00 open"
    ;;

# Symbol Spectrum24 setting (what is the MAC address ???)
*,*,*,XX:XX:XX:*)
    INFO="Symbol Spectrum24 example"
    ESSID="Essid string"
    ;;

# Generic example (decribe all possible settings)
*,*,*,*)
    INFO="Fill with your own settings..."
    # ESSID (extended network name) : My Network, any
    ESSID=""
    # NWID/Domain (cell identifier) : 89AB, 100, off
    NWID=""
    # Operation mode : Ad-Hoc, Managed, Master, Repeater, Secondary, auto
    MODE=""
    # Frequency or channel : 1, 2, 3 (channel) ; 2.422G, 2.46G (frequency)
    FREQ=""
    CHANNEL=""
    # Sensitivity (cell size + roaming speed) : 1, 2, 3 ; -70 (dBm)
    SENS=""
    # Bit rate : auto, 1M, 11M
    RATE=""
    # Encryption key : 4567-89AB-CD, s:password
    KEY=""
```

```
    # RTS threshold : off, 500
    RTS=""
    # Fragmentation threshold : off, 1000
    FRAG=""
    # Other iwconfig parameters : power off, ap 01:23:45:67:89:AB
    IWCONFIG=""
    # iwspy parameters : + 01:23:45:67:89:AB
    IWSPY=""
    # iwpriv parameters : set_port 2, set_histo 50 60
    IWPRIV=""
    ;;
esac
```

/etc/pcmcia/wireless.opts

```
# /etc/network/interfaces -- configuration file for ifup(8), ifdown(8)

# The loopback interface
auto lo
iface lo inet loopback

# The first network card - this entry was created during the Debian
installation
# (network, broadcast and gateway are optional)
auto eth0
iface eth0 inet static
        address 143.160.9.75
        netmask 255.255.252.0
        network 143.160.11.0
        broadcast 143.160.11.255
        gateway 143.160.8.200
auto eth1
iface eth1 inet static
        address 192.168.0.3
        #address 143.160.11.164
        netmask 255.255.255.0
        network 192.168.0.0
        #broadcast 143.160.11.255
        #gateway 0.0.0.0
```

References

[1] O'Shea D., "Wireless LAN: Carriers Draw the Line on Mobile/Wi-Fi Integration", Telephony, vol. 244 no 7, pp. 26, April 2003.

[2] Ylianttila M., Pichna R., Vallstrom J., Makela J., Zahedi A., Krishnamurthy P. and Pahlavan K., "Handover Procedure for Heterogeneous Wireless Networks", Global Communication Conference, 1999.

[3] Ameri A., "Seamless Mobility", [Web:] [http://www.etinium.net/wirelesslan.asp], [Date of access: 20 June 2003].

[4] Pahlavan K., Krishnamurthy P., Hatami A., Ylianttila M., Makela J., Pichna R. and Vallström J., "Handover in Hybrid Mobile Data Networks", IEEE Personal Communications, vol.7, issue 2, pp. 34-47, April 2000.

[5] Lindemann C. and Thümmler A., "Performance Analysis of the General Packet Radio Service", [Web:] http:// rul-www.cs.uni-dortmund.de/publications/ICDCS01.pdf, [Date of access: 5 July 2003].

[6] Kalden R., Meirick I. and Michael M., "Wireless Internet Access Based on GPRS", IEEE Personal Communications, pp. 11-10, April 2000.

[7] Bettester C., Vogel H. and Eberspacher J., "GSM Phase 2 + General Packet Radio Service GPRS: Architecture, Protocols, and Air Interface", [Web:] http:// www.comsoc.org/livepubs/surveys/ public/3q99issue/pdf/Bettstetter.pdf, [Date of access: 9 September 2003].

[8] Karagiannis G. and Heijenk G., "QoS in GPRS", [Web:] http:// doc.utwente.nl/fid/1262, [Date of access: 24 October 2003].

[9] van Rhyn P., "Introduction to Mobile Communications", Quantum Journal, October 2002.

[10] Intelligraphics, "Introduction to IEEE 802.11",
[Web:] http://www.intelligraphics.com/articles/80211_article.html,
[Date of access: 19 August 2003].

[11] Hp, "Understanding WiFi",
[Web:] http://www.peoplesoft-hp.com/SalesTools/whitepapers/%5Bfiles%5D/Mobile/Understanding%20Wireless%20LANs.pdf, [Date of access: 20 November 2003].

[12] Alvarion, "IEEE 802.11 Technical Tutorial",
[Web:] http:// www.alvarion-usa.com/RunTime/Materials/,
[Date of access: 15 February 2004].

[13] Wireless LAN Association, "High-Speed Wireless LAN Options 802.11a and 802.11g",
[Web:] http:// www.wlana.org/pdf/highspeed.pdf,
[Date of access: 9 March 2004].

[14] Wireless LAN Association, "Introduction to WLANs",
[Web:] http://www.wlana.org/learn/intro.pdf, [Date of access: 13 October 2004].

[15] Perkins C., "IP Mobility Support", RFC 2002, October 1996.

[16] Karagiannis G. and Heijenk G., "Mobile IP", Open report no.G 3/0362-FCP NB 102 88 Uen, pp.18-20, July 1999.

[17] ipUnplugged, "Mobility and Mobile IP, Introduction",
[Web:] http:// www.ipunplugged.com/pdf/MobileIPIntro.pdf,
[Date of accessed: 24 July 2004].

[18] Deering S., "ICMP Router Discovery Messages", RFC 1256, September 1991.

[19] Cisco, "Mobile IP", [Web:] http://www.cisco.com/go/mobile_ip/, [Date of access: 5 April 2004].

[20] Chaplin K., "Wireless LANs vs. Wireless WANs", [Web:] http// www.sierrawireless.com/ news/docs/2130273_WWAN_v_WLAN.pdf, [Date of access: 19 July 2004].

[21] Aust S., Proetel D., Könsgen A., Pampu C. and Görg C., "Design Issues of Mobile IP Handovers between General Packet Radio Service (GPRS) Networks and Wireless LAN (WLAN) Systems", Wireless Personal Multimedia Communications, 2002. The 5th International Symposium, vol. 2, pp.868 – 872, 27-30 October 2002.

[22] Perkins C., "IP Mobility Support for IPv4", RFC 3220, January 2002.

[23] Levkowetz H. and Vaarala S., "Mobile IP NAT/NAPT Traversal using UDP Tunneling", [Web:] http://www.ietf.org/Internet-drafts/draft-ietf-mobileip-nat-traversal-02.txt, [Date of access: 9 July 2004].

[24] Daniel S., Calvagna A., "802.11 Mobility Framework Supporting GPRS Handover", Internet Draft, pp. 1-11, July 2003.

[25] Ala-Laurila J., Mikkonen J. and Rinnemaa J., "Wireless LAN access network architecture for mobile operators", IEEE Communications Magazine, vol. 39, no.11, pp. 82-9, November 2001.

[26] Garreis J, "Perfecting Wi-Fi Design", Wireless Review, vol. 20, no 5, pp.19-20, May 2003.

[27] Cisco, "Introduction to Mobile IP", [Web:] http://www.cisco.com/en/US/tech/tk827/tk369/technologies_white_paper09186a00800c9906.shtml, [Date of access: 15 August 2004].

[28] Percins C., "MOBILE NETWORKING THROUGH MOBILE IP", IEEE Internet Computing, pp. 59-60, February 1998.

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

[30] Postel J., "INTERNET CONTROL MESSAGE PROTOCOL", RFC 792, September 1981.

[31] Oehler M. and Glenn R., "HMAC-MD5 IP Authentication with Replay Prevention" RFC 2085, February 1997.

[32] van Sebille T.C., "WLAN-GPRS Roaming", [Web:] http://people.spacelabs.nl/~tomvs/downloads/Report%20---%20WLAN%20-%20GPRS,%20Based%20on%20Mobile%20IP%20(v4)_PUBLIC.pdf, [Date of access: 12 October 2003].

[33] Perkins C., "IP Encapsulation within IP", RFC 2003, October 1996.

[34] Perkins C., "Minimal Encapsulation within IP", RFC 2004, October 1996.

[35] Li T., Hanks S., Meyer D. and Traina P., "Generic Routing Encapsulation (GRE)", RFC 2784, March 2000.

[36] Montenegro G., "Reverse Tunneling for Mobile IP, revised", RFC 3024, January 2001.

[37] Plummer D., "An Ethernet Address Resolution Protocol", RFC 826, November 1982.

[38] Calhoun P. and Perkins C., "Mobile IP Network Access Identifier Extension for IPv4", RFC 2794, March 2000.

[39] Srisuresh P. and Egevang K., "Traditional IP Network Address Translator (Traditional NAT)", RFC 3022, January 2001.

[40] Perkins C. and Johnson D., "Route Optimization in Mobile IP", [Web:] http://www.ietf.org/Internet-drafts/draft-ietf-mobileip-optim-11.txt, [Date of access: 12 July 2003].