# Seamless Mobility

## - SEMO: *A Policy-Based Prototype for Handovers in Heterogeneous Networks*

**Gustav Nyberg**

UMEÅ UNIVERSITY
DEPARTMENT OF COMPUTING SCIENCE
SE-901 87 UMEÅ
SWEDEN

25 January 2006
di00gng@cs.umu.se
Master Thesis in Computing Science 20p
Supervisor at TietoEnator: Lars Linder
Supervisor at LTU: Christer Åhlund
Supervisor at CS-UmU: Thomas Nilsson
Examiner at CS-UmU: Per Lindström

## Abstract

A vision of future wireless networks is coexistence of current existing access networks. The different access networks will be bound together into a single network and the Internet Protocol (IP) will be the glue. To fulfil this vision, one of the most challenging parts is to develop sophisticated policies for vertical handovers. When and in what way the handovers are performed, affects the performance of the mobile services. Traditional handover policies are based on received signal strength (RSS) comparison, but to be able to support Quality of Service (QoS), vertical handover policies have to consider additional parameters such as capacity, utilization cost, power consumption, security and QoS guarantees.

This report presents current research about policy-based vertical handover in heterogeneous networks. Furthermore an analysis, design and implementation of SEMO, a policy-based vertical handover system, is presented. SEMO is compatible to the existing Internet infrastructure and is able to perform soft vertical handovers.



*Keywords*: Seamless mobility, Vertical handover, Heterogeneous networks, Policy-based handovers, Multihomed Mobile IP.

# Acknowledgements

First, I would like to thank my supervisors Lars Linder and Dr. Christer Åhlund for all the support and ideas that they have contributed with throughout this thesis. Their taken time has meant a lot for the outcome of this thesis. I will also like to thank my internal supervisor at Umeå University, Tomas Nilsson, for his help to improve the thesis report and Tomas Röjmyr for his innovative suggestions. Furthermore, I like to thank Jan-Olov Carlsson, Torbjörn Lundmark, Robert Brännström and all others that have been involved in this thesis.

# Table of Contents

# List of Figures

# List of Tables

## Abbreviations

| | |
|---|---|
| 3G | Third Generation |
| 3GPP | Third Generation Partnership Project |
| 4G | Fourth Generation |
| AAO | Active Application Oriented |
| AP | Access Point |
| ARP | Address Resolution Protocol |
| BS | Base Station |
| BER | Bit Error Rate |
| CN | Correspondent Node |
| CoA | Care-of Address |
| D-ITG | Distributed Internet Traffic Generator |
| FA | Foreign Agent |
| FTP | File Transfer Protocol |
| GPRS | General Packet Radio Service |
| GSM | Global System for Mobile communication |
| GUI | Graphical User interface |
| GPS | Global Positioning System |
| HA | Home Agent |
| IEEE | Institute of Electrical and Electronics Engineers |
| IETF | Internet Engineering Task Force |
| ISP | Internet Service Provider |
| IP | Internet Protocol |
| LSS | Location Service Server |
| MAHO | Mobile-Assisted Handover |
| MCHO | Mobile-Controlled Handover |
| M-MIP | Multihomed Mobile IP |
| MN | Mobile Node |
| NIC | Network Interface Card |
| NCHO | Network-Controlled Handover |
| PDA | Personal Digital Assistant |
| QoS | Quality of Service |
| RNL | Relative Network Load |
| RSS | Received Signal Strength |
| RTT | Round Trip Time |
| SEMO | Seamless Mobility Prototype |
| TCP | Transmission Control Protocol |
| UDP | User Datagram Protocol |
| UMTS | Universal Mobile Telecommunications System |
| VoIP | Voice over IP |
| WLAN | Wireless Local Area Network |
| WMAN | Wireless Metropolitan Area Networks |
| WPAN | Wireless Personal Area Networks |
| WWAN | Wireless Wide Area Networks |

# Chapter 1

## 1.1 Introduction

In the new generations of wireless networks, seamless mobility across heterogeneous networks will be supported. A widespread vision of the fourth generation (4G) mobile networks or Next Generation Networks (NGN) includes coexistence of current wireless technologies such as WLAN, General Packet Radio Service (GPRS) and Universal Mobile Telecommunications System (UMTS). Different technologies will be bound together into a single network and the IP will be the glue. The mobile nodes (MN) will be equipped with multiple access networks cards and users will be able to roam transparently over the network in a seamless manner. To realize this vision, much effort has to be put into standardizations, architecture design and access networks' coverage. Another critical aspect is to develop sophisticated policies for deciding when an MN shall handover between the different network interfaces.

A *horizontal* handover is a handover within the same access network technology whereas a *vertical* is a handover between different kinds of access networks [1]. To take the next step to fulfill the vision of a high-quality seamless mobility network, sophisticated vertical handover polices that consider network parameters such as capacity, utilization cost, power consumption, security and QoS guarantees have to be defined.

In this thesis rapport, current research about policy-based vertical handover in heterogeneous networks is presented. Furthermore a design, implementation and performance test of a policy-based vertical handover prototype is presented. The prototype is called SEMO (from *Seamless Mobility*).

### *1.2 Problem Background*

To be able to meet MNs' QoS requirements, the next generations of wireless networks will probably be a multi-access solution. Current existing access networks will coexist in a single heterogeneous network. The main motivation for a coexisting heterogeneous network is the impossibility to define a generic radio access network that can combine optimal bandwidth with optimal coverage at a reasonable cost.

This master thesis is conducted at TietoEnator Telecom Partners AB in Ursviken, in cooperation with Umeå University. The master thesis builds upon a previous master thesis. In that thesis, a prototype based on Mobile IP was implemented. The prototype could perform handovers between network interfaces such as GPRS, UMTS and WLAN. This implementation was preliminary and “prototype-like”. Before starting the prototype all interface connections had to be setup manually and handovers could only occur between pre-known networks. During a handover, the prototype suffered from heavy packet losses.

## *1.3 Problem Statement*

To fulfil a vision of coexistence of current existing access networks, one key challenge is to develop sophisticated policies for vertical handovers. To support high QoS, the policies have to consider network parameters such as capacity, utilization cost and power consumption. A further challenge is that sessions running on an MN must be kept even though the MN changes its point of attachment from one underlying access network to another. AN MN must also be able to automatically discover and establish connections to nearby access networks.

## *1.4 Scope*

The scope of this master thesis is to explore, define, implement and evaluate policies for handovers in heterogeneous networks. The handover mechanism applied by the previous prototype shall be improved, if time and necessity exists. The prototype, SEMO, is bound to be functional in the existing Internet infrastructure.

## *1.5 Aims*

The aim with this master thesis is to implement a prototype application that performs policy-based handovers in heterogeneous networks. The policy should be based on multiple given static and dynamic parameters. All active connections should maintain their connectivity during a handover.

### 1.5.1 Requirements

**Req.1:** A user defined policy for selecting available network interfaces defined on a number of parameters, e.g. cost, signal strength and battery consumption should be implemented.

**Req.2:** If an interface loses contact, reestablishment attempts should be performed automatically.

**Req.3:** A handover between different network interfaces should be performed automatically if current interface loses contact or the policy triggers a handover.

**Req.4:** A handover between different network interfaces should be possible to trigger manually.

### 1.5.2 Characteristic Requirements

**Req.5:** The process for discovering and establish connections to networks should be improved to occur automatically, as this is done manually today.

**Req.6**: Perform and evaluate characteristics measurements should be conducted. Suitable characteristics can be handover time, packet loss during handover and latency in different situations.

### 1.5.3 Implementation Requirements

**Reg.7:** The implementation should be divided into modules with a better defined interface than it is today, to make it easier to add new network technologies later.

**Reg.8:** Platform for the prototype application should be Linux.

**Reg.9:** Implementation language can be chosen freely depending on preference and suitability.

**Reg.10:** Analysis and design documentation as well as user guides and the final report should be written in English.

## *1.6 Methodology*

To fulfil the requirements for this thesis the following steps will be taken: Current research about policies in heterogeneous networks will be explored. The prototype from previous thesis conducted at TietoEnator Telecom Partner AB will be re-designed to fit the requirements for SEMO. Polices, inspired by present research, will be defined and implemented. Measurements regarding handover performance will be conducted and analysed. SEMO will finally be tested and evaluated in different usage scenarios.

## *1.7 Structure of thesis*

This thesis report consists of seven chapters and the structure is as follows: Chapter 2 presents a theoretical overview of important terms, algorithms, policy models and architectures that are commonly utilized in vertical handover systems. The end of the chapter is devoted to experimental solutions and optimizations that can be applied to a vertical handover system. Chapter 3 describes a use-case where SEMO contributes to better network coverage and higher QoS than any existing network technology can offer. Chapter 4 describes and analyzes SEMO´s handover and policy decision model. Chapter 5 gives a system overview by presenting SEMO’s architecture and algorithms. Chapter 6 presents measurements and tests from different handover scenarios. Chapter 7 includes conclusions, limitations and recommended future work.

# Chapter 2

# Theory

This chapter describes important terms, algorithms, policy models and architectures for vertical handover systems. First a short introduction to the subject is given, including common network interfaces, handovers, heterogeneous networks, seamless mobility and Mobile IP. Then policy models for vertical handovers are discussed. Furthermore some widespread algorithms and techniques in vertical handovers are explained. The last part of the chapter presents some experimental solutions and optimizations that have been conducted in vertical handover in heterogeneous networks research.

## *2.1 Theoretical Introduction*

### 2.1.1 Heterogeneous Networks

The definition of the word *heterogeneous* is *"consisting of elements that are not of the same kind or nature*" [2]. As the definition combined with the word *network* implies, a heterogeneous network is a network that consist of multiple networks technologies. A network that administrates different operating systems or computer manufactures, e.g. Windows and UNIX or PC and Macintosh, can also be defined as a heterogeneous network [4]. A *homogeneous network*, in contrast to a heterogeneous, is a network that consists of a single network technology [3]. Today, heterogeneous environments are expanding and mobile devices often have built-in support for multiple network interfaces. Heterogeneous wireless access networks can in general be divided into:

- Wireless personal area networks (WPAN) – Wireless networks that cover a range-limited geographical area. One example of a WPAN is Bluetooth that can provide an ad-hoc wireless network [5].

- Wireless local area networks (WLAN) – WLAN coverage is up to 50-300 meters and is typically used in buildings such as offices or schools. WLAN can provide wireless Ethernet access without expensive infrastructure, e.g. IEEE 802.11b [5].

- Wireless metropolitan area networks (WMAN) – Wireless networks that cover a geographic area such as a city, e.g WiMAX [5].

- Wireless wide area networks (WWAN) – Wireless networks that extends over a large geographical area, e.g. UMTS [6].

### 2.1.2 Fourth Generation

Even though the third generation (3G) has just been introduced to many mobile phone users, 4G is knocking on the door. This new generation is also referred as beyond 3G (B3G) or 3G+ [5]. There are currently two different visions of how 4G will be realized. One vision considers 4G as a single network that combine all the existing wireless access networks. This vision is also referred as the *European vision*. In this vision, mobile users can roam across heterogeneous networks in a seamless manner [7]. Most straightforward is that the IP protocol will be used as the glue between different underlying layers [8]. To fulfill this vision, much effort must be put in standardizations, architecture design and access network coverage [5]. The other vision of 4G is the *linear vision* and this vision is more established in Asia. In this vision current technologies like 3G will be extended so higher data rates can be supported. If this linear vision shall be fulfilled, the capacity of future wireless networks must evolve. Despite differences, both the Asian and the European vision agree on the following key aspects:

- High data rates will be achievable. Rates up to 100 Mbps for wide coverage and 1 Gbps for local coverage will be achievable.

- Both the access and the core network will be All-IP.

- Seamless mobility will be supported.

- The cost for infrastructure will be low, 10 per cent lower than 3G's infrastructure cost.

- Short network latency achieved by short connection and transmission delay [7].

### 2.1.3 Seamless Mobility

Seamless mobility is referred to as the event when all sessions of an MN continue to maintain their connection even as an MN changes its point of attachment [9]. If seamless mobility is supported, an MN can roam across heterogeneous networks and keep its connections active. The underlying heterogeneous network is transparent for the user [5]. Assume a scenario when an MN runs sessions with TCP and UDP connections. If the MN performs a handover to another network, MN will receive a new IP-address. The problem that arises is that the other sides of the active TCP/UDP connections do not know the MN's new IP-address. This results in that all ongoing TCP/UDP connections will break. This problem has to be solved when implementing seamless mobility in a networking layer view [10].

### 2.1.4 Mobile IP

Mobile IP is a proposed and standardized protocol to enable IP mobility [11]. By implementing Mobile IP, a testbed for seamless handover can be provided, without reforming the existing Internet infrastructure. Most Mobile IP testbeds consist of laptops equipped with network technologies that support different access networks such as 802.11b, UMTS and GPRS. The laptops usually run an open source operating system like Linux [5]. Mobile IP consists of the following components:

- **Mobile Node** (MN): The end-system, e.g. a laptop or personal digital assistant (PDA). The MN is free to change its point of attachment in the Internet. The MN has a static IP-address in its *home network*, meaning the subnet that the MN originally belongs to. Any other network than the home network is referred to as a *foreign network*.

- **Correspondent Node** (CN): The CN is the fixed or mobile node that sessions running on the MN communicates with. In a scenario where an MN downloads a file from a File Transfer Protocol (FTP) server, the server is the CN.

- **Care of address** (CoA): The current IP-address for the MN. If the IP-address is a global IP-address, it is said to be a *Co-located* CoA.

- **Home Agent** (HA): An HA is an application running on a computer or router located at the MN's home network. The HA captures IP-packets that belongs to the MN and *tunnels* them to the MN CoA. Tunnelling referrers to the mechanism when intercepted IP-packets are encapsulated in new IP-packets, with headers matching the CoA, and sent to the MN or the *foreign Agent*.

- **Foreign Agent** (FA): An FA is an application running on a computer or router located at a foreign network. If the MN's CoA is not Co-located, an FA has to be implemented at the foreign network. In this case, the HA will tunnel the MN's IP-packets to the FA, which decapsulates them and delivers them to the MN. If the CoA is Co-located, an HA can tunnel the packets directly to the MN.

Fig.1 illustrates the architecture of Mobile IP.

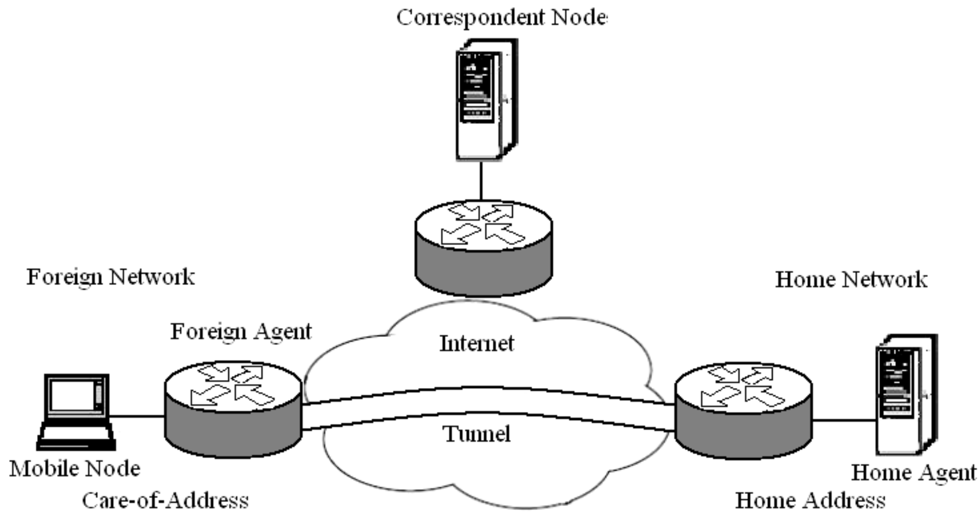


**Figure 1: Architecture of Mobile IP.**

**Scenario:** A CN wants to communicate with the MN and sends therefore IP-packets to the MN's home address. If the MN is away from its home network and has a Co-located CoA, the HA intercepts the packets and tunnels them directly to the MN. If the CoA is not Co-located, the HA will instead tunnel the packets to the FA. The FA will receive the packets, decapsulate them and forward them to the MN. If the MN has a Co-located CoA and wants to send a reply to the CN, it tunnel the packets to the HA. If the CoA is not Co-located, the MN will send the packets to the FA, which tunnels the packets to the HA. When the HA receives the packets, the HA will forward them to the CN [10]. The redirection of packets between the HA and MN is transparent for the CN.

IP-packets that come from a sub network and do not contain a source IP address belonging to that sub network, are usually rejected by routers. This is called *ingress filtering* [12]. Due to ingress filtering, all communication from an MN to a CN has to go through the HA, since one end of their connection is bound to the MN's home address. If Mobile IP is extended with *route optimization*, the HA informs the CN of the MN's CoA. The CN and the MN are now able to communicate directly with each other, bypassing the HA [13]. Mobile IPv4 does not support route optimization.

IPv6 is a new IP version intended to replace the current Internet Protocol IPv4. In IPv6 the address space has increased from IPv4's 32 bits to 128 bits. This leads to much more available IP addresses [14]. IPv6 has a built-in increased support for Mobile IP. Some major differences of Mobile IP in IPv6 contra IPv4 are:

- In IPv6 there will be no need for foreign agents, due to the IPv6 feature *stateless autoconfiguration*. With stateless autoconfiguration servers such as Dynamic Host Configuration Protocol (DHCP) may not be needed. This is because the MN will configure its own global IPv6 address by appending a router supplied fixed local address prefix (64 bit) with its own interface identifier (64 bit). When the MN has ensured uniqueness with duplicate address detection, the outcome is a global 128 bit IPv6-address [15]. The result is that the MN will always have a Co-located CoA and with that the necessity of foreign agents disappears.

- The IPv6 protocol has built in support for route optimization. Route optimization is expected to be used between all MNs and CNs on a global scale.

- In IPv6 both the CoA destination field and the home address destination option field are part of headers. The result is that the packets are allowed to pass through filtering routers [12].

### 2.1.5 Handover

Handover (or handoff) refers to the event when an MN changes its point of attachment from one access network to another. Handovers could either be horizontal or vertical, see fig.2. Handovers is said to be seamless if the handover is transparent to the user of the MN [16]. A definition of a seamless handover is: *"A handover scheme that maintains the connectivity of all applications on the mobile device when the handover occurs"* [1]. A handover to a wireless network with a larger cell size is called an *up-ward handover* and a handover to wireless network with a smaller cell size is a *downward handover* [17]. Handovers can furthermore be categorized as a hard or soft handover. In a hard handover, the MN's network connection is only active through one access network during the handover procedure. This results in a connectivity break in the actual handover moment [5]. In soft handovers there will be no connectivity break. A soft handover could either be achieved by packet forwarding between base stations (BS)/access points (AP) or by letting the MN be connected to more than one access network during a handover [6]. In the second soft handover approach, the access networks connections are not released until the handover is complete. In this approach, the MN has to be connected to multiple networks simultaneously. The MN usually solves this by utilizing multiple networks cards, but nevertheless multiple networks connections via a single networks card can be achieved [18].

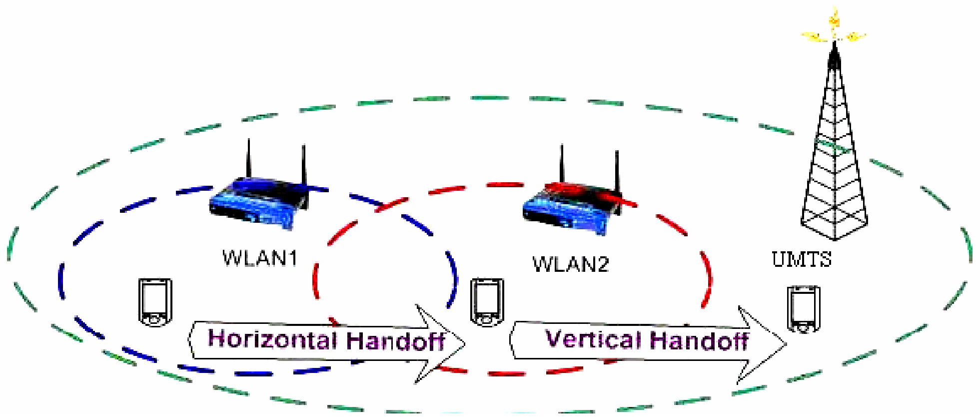


**Figure 2: Horizontal and vertical handover [1]**

Traditional handover polices are based on RSS comparisons. In homogeneous network RSS policies may be sufficient, but in heterogeneous network these policies are not as reliable and correct. This is due to the fact that scenarios may exist where for example a network interface such as WLAN may have low signal strength but still is capable of offering higher QoS than a GPRS interface with currently higher signal strength [17]. To fulfill the vision of a flexible and high-quality vertical handover system, algorithms have to be more sophisticated. The decision algorithm for choosing the best access network has to consider users' requirements for bandwidth, cost, power consumption and the current context such as security and QoS guarantees [6]. Modern research in heterogeneous networks focuses increasingly on finding efficient policies for appointing the most suitable access network.

### 2.1.6 Network Interfaces

In this report's context, the term *interface* refers to different wireless access networks. Table 1 presents the most common and wide spread interfaces that are available on the market today. Worth mentioning is that most implemented tests and experiments from the reference literature have used IEEE 802.11b, UMTS and/or GPRS.

**Table 1: Brief presentation of wireless technologies [19] [10].**

| Network | Description | Date rate | Coverage |
|---|---|---|---|
| Satellite | Satellite support for mobile communication. | Up to 144 Kbps | Continental |
| GPRS (2.5G) | Always-on characteristic, packet-oriented, can be used parallel to GSM (Global System for Mobile communication) services. | Depending of the coding and free time-slots, rates up to 170 Kbps. | Approx. 35 Km |

| | | | |
|---|---|---|---|
| EDGE (Enhanced Data rates for GSM Evolution) | Used over the GSM network and can provide up to three times the data capacity of GPRS. | Up to 384 Kbps | Approx. 35 Km |
| UMTS (3G) | Always-on characteristic, packet-oriented, uses *Wideband Code Division Multiple Access* (W-CDMA) for separate users to access to medium, defined by 3rd generation partnership project (3GPP). | Up to 2 Mbps | 20 Km |
| HSDPA (High Speed Downlink Packet Access) | A mobile telephony protocol (3.5G) that uses a new W-CDMA standard. Available in Europe earliest 2006. | Up to 8-10 Mbps (downlink) | Up to 5 Km |
| IEEE 802.11b | The most widespread WLAN standard. MNs connect to an access point. Radio transmits at 2.4 GHz. | 1, 2, 5.5 or 11Mbps, depending of distance from the sender and receiver. | 50 - 300 m |
| IEEE 802.11a | Radio transmits at 5 GHz. Higher performances in close range than 802.11b. | Up to 54 Mbps | 50 - 300 m |
| IEEE 802.16 (WiMAX) | A high capacity wireless broad band on Metropolitan scale. | Up to 70 Mbps | 50 Km |
| Bluetooth | Provide range-limited wireless services with no need for infrastructure (ad-hoc net). | Max. 1 Mbps | 10/100 m |

### 2.1.7 QoS in Wireless Networks

The physical channel of wireless networks is unpredictable and their frequency spectrum is often limited. Different network technologies have to use these resources efficiently to be able to support the QoS requirements from applications running on an MN. To be capable to provide services such as streaming video, wireless network traffic can be divided into different service classes. For example, UMTS has four service traffic classes: *conversational*, *streaming*, *interactive* and *background classes*. The conversational traffic, e.g. video telephony, has the highest priority and the background classes, e.g. email, has the lowest [20]. The IEEE 802.16

standard, WiMAX, also has four different service classes: *Unsolicited Grant Service*, *Real-time Polling Service*, *Non-real-time Polling Service* and *Best Effort* [21]. In general, the IEEE 802.11 offers poor support of QoS capabilities, but the upcoming 802.11e standard offers different traffic categories and with that significant improvement to e.g. multimedia traffic in WLANs [22]. Depending of their tolerance against bandwidth variations, applications can be divided into *elastic* or *inelastic* applications [20]. The most commonly used QoS parameters that wireless network interfaces provide to an application are:

- Bandwidth: Bandwidth is amount of data that can be transmitted during a fixed time, often expressed in bits or bytes per second [23]. Real-time applications such as video streaming are often elastic applications. Compared to elastic applications, inelastic applications are less tolerant to variation in bandwidth. The bandwidth parameter is considered to be one of the most important.

- Delay: Delay is the amount of time taken from a transfer from one node to another. Network Interactive applications such as network games often demands short delays.

- Jitter: Jitter is the over time variations in delay. Jitter intolerant applications such as real-time applications do often use different buffering techniques to compensate for jitter [20].

- Bit Error Rate (BER): BER is defined as the amount of errors per total number of transferred bits [24]. The bit error rate is much higher in wireless networks then in wired networks [20].

## 2.2 Policies

The term *policy* is defined in different ways by different literature. However, all literature agrees that policies are some sort of rules. Here are some proposed definitions:

- “Policy is a rule that describes actions to be taken when certain conditions happen” [25]**;**

- “The combination of rules and services where rules define the criteria for resource access and usage” [26];

- “Policies are rules governing the choices in the behavior of the system” [27];

The XML-based Intermediary Rule Markup Language (IRML) was proposed in an Internet Engineering Task Force (IETF) internet draft to be the language that should be used for specifying policies [28]. The purpose of having policies specified in a certain language is to enable the possibility to download policies in run-time [25]. In 1999, the first policy-enabled handovers system in heterogeneous networks was presented. Traditionally handover decisions in heterogeneous networks are based on RSS, but more sophisticated policies have to be developed to support seamless mobility [29]. Sophisticated policies must be context-aware, meaning that they monitor the MN's state and surroundings. The context information affects the outcome of the policy decisions [30]. Suggested context parameters that should be considered are:

- *User context*: The user profile and preferences such as specification regarding requested bandwidth, money willing to spend or the user's need of long lasting battery power. The user's geographic location, behaviour pattern or moving speed can also be used.

- *Network context*: Available interfaces coverage, bandwidth, latency, traffic load, packet loss, monetary cost or security aspects can be used as policy parameters concerning the networks context.

- *Mobile Node context*: The MN's operating system configurations, power status etc.

- *Application context*: In the application context parameters as which protocols the current application uses, the application's bandwidth requirement, the application's duration etc. may be considered.

Policy parameters can either be static or dynamic. Typically static parameters are user preference and the cost of the different access networks, whereas the user's moving speed, the MN's power status and RSS are typically dynamic parameters. The different parameters can be obtained through periodically monitoring, measurements and by statistical analysis [31]. The utilization of sophisticated polices will increase the complexity of the handover decision process. Optimized architectures and well designed evaluation functions are needed to speed up the handover decision so user satisfaction can be obtained [23].

## 2.3 Policy model

The ideas and terms of an IETF proposed policy model [26] for the handover decision making process in heterogeneous networks, are widely spread. Fig.3 illustrates a policy model that is based on the IETF proposed model.

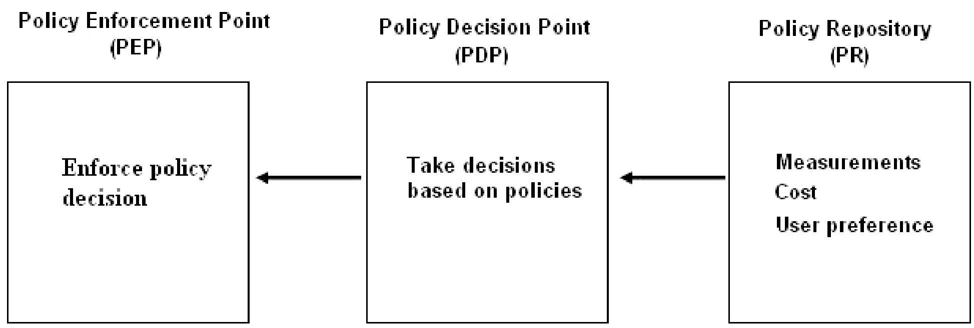


**Figure 3: Policy-Based Decision Model.**

As fig.3 shows, the model consists of three different entities:

- A Policy Repository: The Policy Repository (PR) is responsible to deliver requested policy parameters to the Policy Decision Point (PDP). The PR contains information such as user preferences, the signal strength and cost of available access networks. The PR can obtain information through measurements of the environment.

- A Policy Decision Point: The PDP is the control entity that evaluates access networks through policy decisions. The policy decisions are based on the parameters received from the PR. If the PDP decides that a handover is motivated, the PDP informs the Policy Enforcement Point (PEP) to perform a handover.

- A Policy Enforcement Point: The PEP receives policy decisions from the PDP. It is the PEP that actually performs the handover. The PEP is said to enforce the policy decision [32].

The physical location of the PDP separates policy systems to perform a mobile-controlled handover (MCHO) or a network-controlled handover (NCHO) [6]. The MN makes the handover decisions in a MCHO system whereas the network makes handover decision in a NCHO system. A mobile-assistant handover (MAHO) is when measurements are done by the MN and reported to the network, which in turn makes the handover decisions [6].

In a NCHO/MAHO policy system, the PDP is located in a network node [19]. GSM is one example of a MAHO-system, where the measurements are done at the MN and sent to the BS/ Mobile Services switching Centre (MSC), which makes the handover decisions [5]. If the PDP and the PEP are not located at the same place, an IETF internet draft [28] suggests that *Common Open Policy Service* (COPS) protocol should be used for their communication. The advantage with implementing the PDP at a network server is the availability of network information such as capacity, coverage, mobility support and current load. This gives the PDP full knowledge of the network capacity, and it can therefore avoid overloaded networks. By applying a NCHO/MAHO approach, a greater level of QoS can be offered to a general user, compared to if the MN itself can utilize any requested access network [33].

In a MCHO policy system, the PDP is located at the MN. The local decision depends on information such as battery level and other conditions at the particular MN [26]. Vertical handover systems are generally designed as MCHO systems [8]. One advantage with MCHO is that the MN is able to determine the exact time to perform the handover. This is due to the fact that the MN has full control over its own context, e.g. which applications that are running, state of transport connections [19]. When using a NCHO or MAHO, a wireless protocol for sending information from/to the MN must be used. In a MCHO, less information has to be sent between the MN and the network. This argues for a MCHO system, because it is not efficient to rely on the wireless environment [25].

## *2.4 Common policy algorithms*

This section describes common policy algorithms. The term *cost function* is described and clarified by an example. Furthermore, the *hysteresis margin* and the *dwell-timer* algorithm are presented. Both algorithms are used to stabilize vertical handover systems.

### 2.4.1 Cost function

Cost functions (or score functions) defines rules for making optimal handover decisions [5]. In 1999, Helen Wang [27] was first out to present a policy-based handover system. The system used a cost function to evaluate available access networks. The cost function took user preferences in consideration and calculates a policy value for each network. The network interface that receives the lowest policy value was considered as the most suitable access network to utilize. The cost function was preliminary and could not handle all scenarios e.g. if an access network was free of charge. In 2004, researchers at the University of California, Los Angeles (UCLA) presented another cost function [1]. This cost function will now be described in detail, with the purpose to clarify how a cost function is used, whereas the basic idea of most cost functions is the same. This cost function was implemented on the seamless handover architecture called *Universal Seamless Handover Architecture*, but can be applied on any vertical handover architecture. This cost function, like Wang’s, is used to calculate a policy value for access networks. The network with the highest value is considered as the best interface. If a handover is motivated, the network with the highest value becomes the handover target [1]. The proposed cost function, $S_{i,}$ is shown below in function 1.

$$S_i = \sum_{j=1}^{k} w_j f_{j,i} \qquad 0 < S_i < 1 \qquad \sum_{j=1}^{k} w_j = 1 \qquad (1)$$

The normalized policy value for network interface *i*, $S_{i,}$ are the sum of *k* weighted functions $f_{j,i}$. The weights, $W_n$, are calibrated from user requirements. The sum of all weights is always 1. The normalized functions $f_{j,i}$ are used to provide a score value for network interface *i*'s parameter *j*. Parameter *j* can for example be the *interface i'*s monetary expense (E), link capacity (C) or power consumption (P) [1]. If the mentioned factors are used, the score function will look like function 2 illustrates:

$$S_i = w_e f_{e,i} + w_c f_{c,i} + w_p f_{p,i} \qquad (2)$$

The normalized functions for monetary expense (E), link capacity (C) and power consumption (P) are illustrated below. The following constrains must be fulfilled for the introduced coefficients: $a_i \geq 0$, $M \geq \beta_i$ and $\gamma_i \geq 0$.

$$f_{e,i} = \frac{1}{e^{a_i}} \quad f_{c,i} = \frac{e^{\beta_i}}{e^{M}} \qquad f_{p,i} = \frac{1}{e^{\gamma_i}}$$

The values for coefficient $\alpha_i$, $\beta_i$ and $\gamma_i$ can be obtained from a look-up table or a well-tuned function. The coefficient *M* is the maximum demanded bandwidth requirements from the user. Here is a scenario that shows how the cost function is used to appoint the best available access network. The following parameters for network interface *i* are used: the monetary expense *x*, the measured link capacity *y* and the network's power consumption *z*. The coefficients $\alpha_i$, $\beta_i$ and $\gamma_i$ are assumed to be:

$$a_i = \frac{x_i}{20} \quad \beta_i = \frac{Min(y_i, M)}{M} \quad \gamma_i = \frac{2}{z_i}$$

**Scenario:** A user is carrying an MN, which has a vertical handover application running and currently utilizes UMTS for internet connectivity. As the user enters an internet café, the vertical handover application discovers the café's 802.11b network. The cost function will now calculate a policy value for the 802.11b network, to consider if a handover is motivated. The user is concerned with his/hers battery consumption, so he/she has added a little extra weight on the power consumption ($W_p = 0.4$). The user's desire for a low monetary expense and high capacity are equally matched ($W_e = W_c = 0.3$). Table 2 presents the network parameters that are used in this scenario.

**Table 2: Assumed network parameters. *The cost/min is based on a monthly subscription including 1 gigabyte capacity usage [34].**

| Interface | Cost/min | Capacity | Battery lifetime | Max demanded bandwidth |
|---|---|---|---|---|
| 802.11b | 0.50 kr | 5 Mbit/s | 2 hours | 2 Mbit/s |
| UMTS | 0.09 kr* | 300 kbit/s | 4 hours | 2 Mbit/s |

Based on the parameters from table 2, the cost function calculates policy values from both the 802.11b and the UMTS network interface, see fig.4. The value for the 802.11b network is 0.55 and the UMTS's policy value is 0.59. This means that the UMTS network is considered as a better match to the user's QoS requirements. The vertical handover system decides that no handover is motivated and the MN continues to utilize the UMTS access network.

$$
\begin{aligned}
S_{802.11b} &= 0.3\times\frac{1}{e^{\alpha}} + 0.3\times\frac{e^{\beta}}{e^{M}} + 0.4\times\frac{1}{e^{\gamma}} \\
&= 0.3\times\frac{1}{e^{0.5/20}} + 0.3\times\frac{e^{2/2}}{e^{2}} + 0.4\times\frac{1}{e^{2/2}} \approx 0.55 \\
S_{UMTS} &= 0.3\times\frac{1}{e^{\alpha}} + 0.3\times\frac{e^{\beta}}{e^{M}} + 0.4\times\frac{1}{e^{\gamma}} \\
&= 0.3\times\frac{1}{e^{0.09/20}} + 0.3\times\frac{e^{0.3/2}}{e^{2}} + 0.4\times\frac{1}{e^{2/4}} \approx 0.59
\end{aligned}
$$

**Figure 4: Cost function example for UMTS and 802.11b network.**

## 2.4.2 Hysteresis

A hysteresis margin can be introduced to increase the stability in a handover system. Worth mentioning is that hysteresis margins are not a complete handover solution. When comparing access networks' QoS levels, e.g. through cost function values, a hysteresis margin is added to the current access network's QoS level value. With this added value, handover only occurs if another network has a profitably higher QoS level. The hysteresis margin contributes to avoid "ping-pong effect" and oscillation between the current network and another network with about equivalent QoS level [35]. An added hysteresis can look like this:

$$\textit{Handover when: } QoSLevel_{New} > (QoSLevel_{Current} + Hysteresis)$$

When roaming across heterogeneous networks, performance can significantly improve with an optimally determined hysteresis value, due to fewer unmotivated handovers [36].

### 2.4.3 Dwell-timer

A Dwell-timer algorithm is used to ensure that a potential handover target is a stable network. The algorithm is often used in combination with a hysteresis margin. Dwell-timers are not complete handover solutions and are as hysteresis margins used to optimize an existing handover system [5]. The basic idea with dwell-timers is to for a certain time resist a new access network with a high QoS level. If the network maintains the high measured QoS level as long as the dwell timer countdown, the network is categorized as stable. If the new network is both stable and can provide a higher QoS level than the currently used, it is seen as a suitable handover target. If the measured network could not maintain the high QoS level, the network is classified as unstable, and therefore not a suitable handover target. The number of users, weather conditions or traffic can cause networks' QoS level to rapidly decrease [37]. Fig.5 illustrates a simple dwell timer algorithm:

1. Measure the QoS level of all available interfaces
2. If QoS $level_{other}$ < QoS $level_{current}$ + Hysteresis
    - 2.1 Go back to 1
3. Else
    - 3.1 Start dwell-timer
    - 3.2 If condition endures until timer expires
        - 3.2.1 Handover the new interface
    - 3.3 Else
        - 3.3.1 Stay with current interface

**Figure 5: A dwell-timer algorithm.**

For an optimized handover, the best countdown time for the dwell-timer must be analysed. A dwell-timer based handover algorithm has shown significantly better results, in comparison with a RSS algorithm, when the MN travels at higher velocities [16].

## *2.5 Related research*

### 2.5.1 Battery aspects

For mobility reasons, an MN must rely on battery power and can only operate as long as the battery allows it. Low battery consumption is therefore a very important aspect for an MN. Despite massive research efforts, the development of longer lasting batteries advances slowly. Promising experiments have been conducted with fuel cell technology [38], that offer ten times better capacity than current batteries. Unfortunately, these fuel cell batteries are a long way from being put on the market [25]. Today most vertical handovers solutions are not using an effective power management. The most common thing is that every network interface is active all the time. The MN usually receives or sends network measurements information, even when the sessions on the MN QoS requirements are fulfilled. Handover systems with this design tend to have high power consumption and unnecessary packet delivering is also considered as waste of bandwidth. Some handover systems are designed so that interfaces are only periodically activated. This approach is more battery effective than the always-on scenario. Nevertheless, unnecessary packets will be sent/received and network interfaces will be activated even when they are not needed [37].

Different algorithms have been proposed too decrease the MN's battery consumption. One proposed algorithm is the *Active Application Oriented* (AAO) scheme [37]. The main idea with the AAO is to only activate another access interface, than the current interface, when a handover is motivated. A *Location Service Server* (LSS) is introduced to provide the MN with information about nearby network interfaces. The information that the MN receives from the LSS can include network's parameters such as available bandwidth and coverage area. The MN is proposed to use Global Positioning System (GPS) to obtain its own geographical position. The AAO scheme consists of two decision algorithms.

The first algorithm, see fig.6, is executed every time an MN boots up or changes mode, e.g. from idle mode to video-streaming mode. This algorithm first checks if the current interface satisfies the MN's QoS requirements. If the requirements are fulfilled, the MN is content with the currently utilized network and finds no need to explore other network. If the current network does not satisfy the QoS requirements, network information received from the LSS via the currently used BS, are used in a cost function to calculate a QoS level for every available access network. If a nearby network has a higher QoS level than the currently used, a dwell timer is started to ensure the stability for the new network. If stability is ensured, a handover is performed. If not, the MN continues to utilise the current network.

Executed when MN boots up or change mode.

1. Check if the QoS requirements are satisfied or not.
2. If satisfied
    - 2.1 Stay with current network
3. Else
    - 3.1 Request networks' information for nearby BS
    - 3.2 If nearby networks exist
        - 3.2.1 Calculate $QoS_{nearby}$ and $QoS_{current}$
        - 3.2.2 If $QoS_{nearby} > QoS_{current}$
            - 3.2.2.1 Start dwell timer
        - 3.2.3 If conditions endures until timer expires
            - 3.2.3.1 Handover to nearby network
        - 3.2.4 Else
            - 3.2.4.1 Stay with current network
    - 3.3 Else
        - 3.3.1 Stay with current network
4. Stay with current network

**Figure 6: The first algorithm used by the AAO scheme [37].**

A currently used network that fulfills the QoS requirement will continuously be utilized, until its signal strength becomes too weak. If that happens, AAO scheme's second algorithm will be executed. The second algorithm first tries to perform a horizontal handover to another network with stronger signal strength. If that fails, a cost function appoints the most suitable interface, regarding technology, and performs a vertical handover.

Experimental tests with the AAO show that the scheme has positive effects on the power consumption. In fig.7, an MN with the battery capacity of 1000 mAh was able to operate 10 hours when the "network interfaces always-on"-schema where used. An 802.11b and a GSM/GPRS access network were used in this experiment. If the network interfaces where periodically activated, the MN could operate for 20 hours. Finally, when the AAO scheme where used, the MN could operate for 24 hours [37].

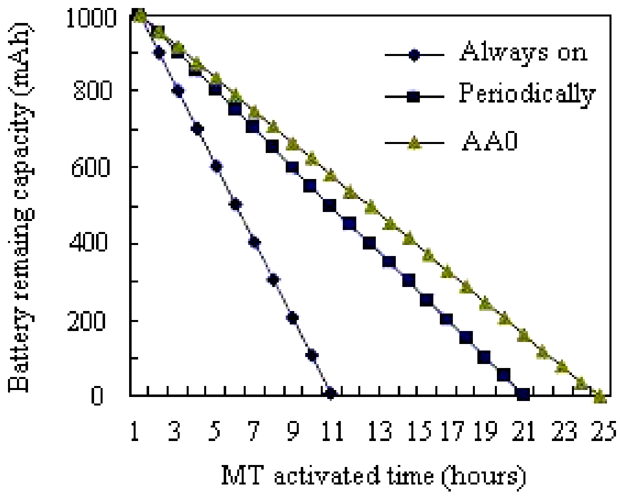


**Figure 7: A mobile node's battery life-time when using "interface always-on", periodically activated or AAO scheme [37].**

Paper [16] also proposes a solution for handover system with the goal to decrease the MN's battery consumption. Assume a scenario where cellular cell-sizes, UMTS or GPRS, are geographically much larger than WLAN cells. If an MN moves across such a cellular area and at the same time performs active scanning for a WLAN interface, either by an always-on or a periodically active approach, the MN will not likely encounter WLAN coverage. In this scenario, the MN wastes battery capacity for no use. The authors of paper [16] propose that the MN should receive information of WLAN cells locations from a base station controller and obtains its own geographical position via GPS. The MN should use the location information in combination with its own position, and anticipates the upcoming of a WLAN. By this approach, the MN only has to scan for WLAN cells when they are nearby, which will lead to less battery consumption [16].

### 2.5.2 Network and Mobile controlled handover system

Designing a handover system as either a NCHO/MAHO system or MCHO system has its advantages. The main advantage with a NCHO/MAHO approach is that network information is available for the PDP. When a handover decision is made, the overall network load can be better balanced, in comparison to a MCHO system. One advantage with a MCHO approach is that it gives the PDP better possibilities to exactly determine the best time to perform the handover, due to better knowledge of the active sessions running on the MN [19]. Murray [32] has presented a handover system that is a combination of a MCHO and a NCHO system. The main goal with this proposed system is to optimize the overall network capacity and QoS.

In Murray's handover solution, a PDP is implemented in a location where network information is available, e.g. in a GPRS base station controller, UMTS radio network controller or in a WLAN's gateway router. The policy repository (PR) provides the PDP with information of the network's available bandwidth and link level QoS parameter for active connections. The PDP is responsible for selecting an interface when a new MN arrives to the network or when a known MN has sent a handover request. The PDP keeps track of how many users every cell in the network can support. This is done through constant link level report from different cells. A network health value between 0 and 1 for each cell is also calculated and used when decision are made to admit or not admit a new user to a cell. When a new MN arrives or requests a handover, the PDP admits the MN to the least loaded network. By applying this method, load balance can be achieved [32].

Furthermore, every MN has its own PDP. This PDP has access to link level statistics, user performance, and minimum QoS requirement for active sessions. All this information is stored in the MN's PR. If the current network interface's QoS level falls below a threshold, the PDP will decide whether to send a handover request to the network server or not. If the requested network is not full, the network server's PDP will, as mentioned above, handle the handover request by admitting the MN to the least loaded network cell [32].

### 2.5.3 Multihomed Mobile IP

A node with several points of attachment, multiple IP-addresses, is said to be a multihomed node. A multihomed node can either be a router, a fixed or a mobile node. Multihomed Mobile IP (M-MIP) [40] is an extension of the standard Mobile IP. M-MIP allows an MN to register multiple Care-of Addresses (CoA) and bind them to a single home address. With M-MIP, the MN is able to simultaneously utilize different network interfaces. This feature is desirable in vertical handover over heterogeneous networks because it can contribute to:

- Smooth handovers – In a handover, the connection to the currently used network interface can be kept until the handover target interface has successfully established its connections. By this approach, packet loss can be eliminated [42].

- Stability – If the MN's current interface unexpectedly looses connection, a second already active interface can directly be used as backup. This makes the system reliable in aspect of maintenance of Internet connectivity [41].

- QoS – Different network interfaces can offer different QoS guarantees in perspective of latency, cost, bandwidth etc. The MN can use one interface for a certain session and another interface for a second session, e.g GPRS can be used for Telnet and WLAN for FTP [31]. Another possible scenario is that an MN wants to send packets over all available interfaces. In this scenario, a policy has to be used to divide packet flows over the different interfaces [31].

## 2.6 Comments

*The policy model:* The proposed policy model in [26] includes a Policy Repository (PR), a Policy Decision Point (PDP) and a Policy Enforcement Point (PEP). This model has been accepted and adopted by almost all authors to recent written papers concerning policies in heterogeneous networks. The policy model consists of three different entities: The PR that measure the MN environment, the PDP that makes the policy decision and the PEP that enforces the decision. I do not think that the policy model itself is such a significant discovery, but it has definitely contributed to more unity in the research area. The fact that all papers use the same terms makes it easier for readers.

*Network-controlled handover (NCHO)/ Mobile-assistant handover (MAHO):* In a NCHO and MAHO the network is responsible for making the policy decision. Many papers mention the advantages with a NCHO/MAHO solution and for example [32] point out that the overall network load can be better balanced. I think that the biggest disadvantage for a NHCO/MAHO system is the fact that it does not exist any "network" in existing Internet infrastructure. It does not exist any station that keeps track of available network interfaces, as it does in homogenous networks, e.g. the Radio Network Controller (RNC) in GRPS/UMTS. To make such a station possible, a network provider has to own and control several access networks in the same area. If this is not the case, nobody will have access to necessary network information from the different networks. A possible solution for this problem could be that different network providers cooperated and provided a cooperative network server. If network stations would exist, NCHO/MAHO policies would probably be more developed, but I think that this server is far from reality. Furthermore I think a NCHO/MAHO is only necessary if there are many MNs in the area, because otherwise the direct need for effective load balance decreases. In a MAHO solution, the MN is required to report context measurements to the network station, as done in the GSM network. In heterogeneous networks, this leads to the question regarding who should pay for this extra traffic. I do not think that the users would appreciate to pay for this type of traffic.

*Mobile-controlled handover (MCHO):* In a MCHO the MN is responsible for making the policy decisions. Most conducted experiments and publications regarding policies in vertical handovers, e.g. [19], [25], [43], promote a MCHO decision model. In the existing Internet infrastructure a

MCHO is easier to implement than a NCHO/MAHO. This is because the MN takes its own policy decisions without having to include any none existing network station. I think that the main reason that it does not exist any widespread products within seamless mobility is the need of support, often in form of a Home Agent. I think it is unwise to introduce even more required support, in form of the network server as a NCHO/MAHO system demands, and therefore I favour a MCHO based handover system. Furthermore, I think that papers that "introduce a centralized station" like [32], [16] and [43] do not take full responsibility for their proposed solutions. None of the mentioned papers describes how this "centralized station" would be introduced in reality.

*Battery aspects:* Power consumption is an important aspect for an MN and research has been conducted how to decrease the power consumption in vertical handover systems. Paper [37] and [16] present two different battery saving solutions, see section 2.4.1. Both solutions demand a server that keeps track of available access networks' positions and coverage range and that the MN should obtain its geographical position via GPS. In these papers, the GPS' power consumption is not included, which can be seen as a negative aspect. Paper [37] suggests that the MN's network interfaces should only active as little as possible, to decrease the power consumption. This proposed suggestion included a server that should provide the MN with information of nearby networks, and I think that this server is hard to realize in reality.

*Cost functions:* Almost every policy-based handover system use some sort of cost function in its handover decision process. Paper [1] and [27] present two cost functions that are both used to calculate a policy values for access networks. The network with the highest (or lowest) policy value can be appointed as the "best" interface. I think that it is important that cost function parameters are relevant and can be obtained with realistic effort. Furthermore, I believe that people, especially teenagers, tend to be willing to give up QoS requirements if the price is "right". Therefore, I think that the cost parameter often should put some extra weight to the total policy value.

*Multihomed Mobile IP (M-MIP):* By using M-MIP an MN may utilize many different network interfaces simultaneously. This feature can contribute to smooth and stable handovers. Different applications can furthermore utilize different interfaces for sending traffic. The applications can pick and use the currently available network that is most suitable for QoS requirements of the applications. I think that it can be wise to implement M-MIP if requirements for stable and smooth handovers exist. However, I also think that it is important not to let simultaneous utilization of network interfaces affect the cost and the MN's power consumption too much.

# Chapter 3

# Use-case

## *3.1 The patrolling police officer*

Many predict that coexistence of access networks will be supported by the next generations of mobile networks. The technical know-how is already here, but available products are still very few. To inspire and clarify opportunities, a use-case is going to be described where the prototype from this thesis, SEMO, contributes to better network coverage and higher QoS than any existing network technology can offer:

Today it is common that crowded areas such as centers of cities or airports offers WLAN coverage. The police tend to frequently patrol crowded and well visited regions. Patrolling police officers have a very high need of always being connected with the central police station or with other officers. The police officers currently use mobile phones or communication radios as communication devices, but could in the future instead be equipped with a PDA or another small mobile device. SEMO could be installed on the policemen's communication device. To offer seamless mobility, SEMO must have a computer that runs a home-agent application. This home-agent computer can be stationed at the police station.

When the patrolling police officer leaves the station, his/hers connection to the station's WLAN will start to fade away. At this time, SEMO will automatically establish a UMTS connection and the officer continues his/hers audio and video communication with his/hers colleagues through the UMTS network**.** By using SEMO, the patrolling police officer would automatically utilize the currently best access network, without any interaction from the policeman. The fact that network selection is performed automatically is important for a police who in his job must keep his/hers eyes and arms focused on other things. Furthermore, the policemen themselves do not need to have any knowledge of which QoS parameters available networks offer. In this use-case, security is crucial. A secure communication can be achieved by cryptating the traffic between the station and the police officer**.** If the police officer patrols within a WLAN hotspot, SEMO can contribute to lower cost and higher available bandwidth for the ongoing communication. With higher bandwidth, the policeman would be able to rapidly download large amounts of data and information. By getting this opportunity, the entire efficiency of a future patrolling policeman could evolve.

Other thinkable use-case for SEMO is if a doctor has to make an emergency visit to a remote patient and needs to download medical journals. Due to

SEMO, the medical journals could be downloaded during the transport, via both WLAN-hotspots and UMTS. Another use-case is if a house and a nearby home office both have WLAN coverage. By utilizing SEMO, all connections such as ongoing downloads and voice over IP (VoIP) applications could maintain their connectivity when a person moves between the two buildings. The home office scenario is illustrated in fig.8.

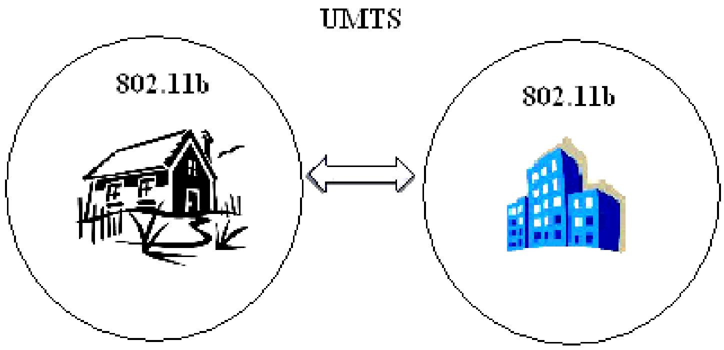


**Figure 8: Home office scenario**

In fact, SEMO can be profitably used in almost every situation where there are many available access networks and the need for ongoing connectivity exists. To summarize, some advantages for using SEMO are:

- Better network coverage can be achieved, e.g. WLAN can offer coverage in building where UMTS coverage fails.

- Higher QoS for the user can be granted, e.g. in form of higher network capacity.

- SEMO can contribute to lower costs. For example, SEMO can help a user to better utilize his/hers monthly private WLAN subscriptions, and with that reduce his/hers mobile phone cost.

- SEMO guarantees that the user will use the currently most suitable network interface. This is an advantage when the user is busy with other things and does not have time to manually change network connections. Furthermore the user does not need to have any own knowledge of which QoS parameters an available access network provides.

- Save resources, battery etc.

# Chapter 4

# Analysis and Design

## 4.1 Seamless mobility

SEMO uses M-MIP to support seamless mobility and consists therefore of an MN and an HA application. SEMO uses the extended M-MIP to be able to register multiple IP-addresses at the HA. This is needed by SEMO's handover mechanism, see section 4.2. SEMO is required to be compatible to the existing Internet infrastructure, in where there exist no FAs. If FA solution should be supported, an FA application must run on the MN's current network. Due to this fact, SEMO does not use any FA and the HA tunnels the packets directly to the MN. By excluding FAs, SEMO can only be used on networks that use public IP-addresses.

## 4.2 Handover mechanism

In a handover, it is desirable to minimize packet loss. The previous prototype, used a single tunnel to encapsulate the traffic between the MN and the HA. A handover from an 802.11b network to a UMTS network with the previous prototype is illustrated in fig.9. In this scenario, the tunnel is initially associated with the 802.11b network. The handover mechanism will perform a handover by reconfiguring the tunnel to be associated with the UMTS network. To reconfigure the tunnel, the tunnel is restarted with new configurations. This handover is classified as a hard handover and will result in a total packet loss during the time it takes to tear, reconfigure and restart the tunnel. This is shown in a handover experiment in section 6.2.1.

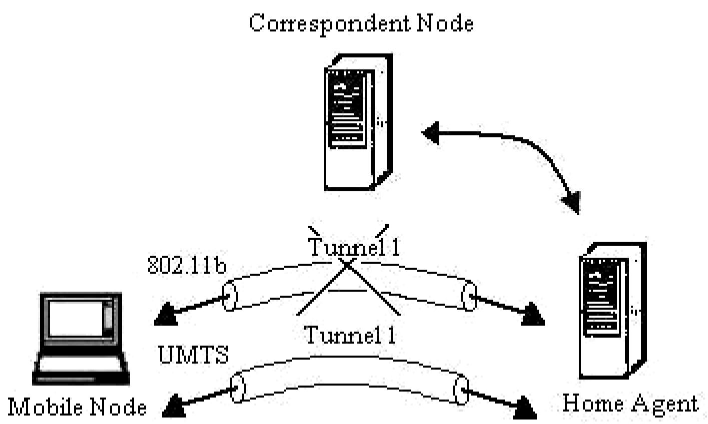


**Figure 9: Hard handover from an 802.11b network to a UMTS network.**

With the ambition to minimize packet loss, SEMO uses a more sophisticated handover mechanism. When SEMO discovers a new network, a tunnel is created and bound to the new network. The created tunnel is kept as long as connectivity to the network exists. By pre-building the tunnel, it is already up and ready to be used in a handover. Due to this design, SEMO is able to perform soft handovers. Fig.10 shows a scenario where both an 802.11b and a UMTS network are accessible. In the figure, the 802.11b network is the currently utilized network.

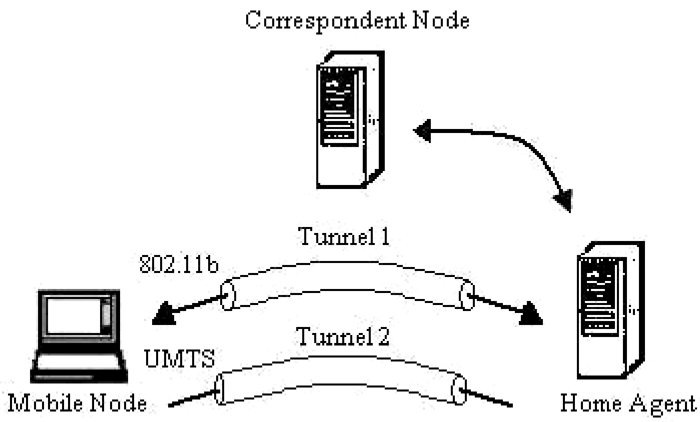


**Figure 10: One tunnel associated with each network interface.**

By establish multiple tunnels, SEMO will not lose packets in a handover between two networks with durable coverage, see section 6.2.2. This is because even when the MN/HA sends traffic over different tunnels all packets, regardless of tunnel, are captured by the HA/MN. In other words, both sides are able to listen and capture packets regardless of which tunnel they arrive from. In a handover between e.g. 802.11b and UMTS, the traffic will be redirected from the tunnel associated with the 802.11b interface to the tunnel associated with the UMTS interface. The redirection of traffic is done through changes in routing rules. Fig.11 shows a scenario where the MN and the HA sends traffic over different tunnels. This scenario might occur when the MN's and HA's redirection of traffic is not synchronised. As mentioned, even though traffic is sent over different tunnels, no packets are lost, due to the fact that SEMO listens to all tunnels simultaneously.

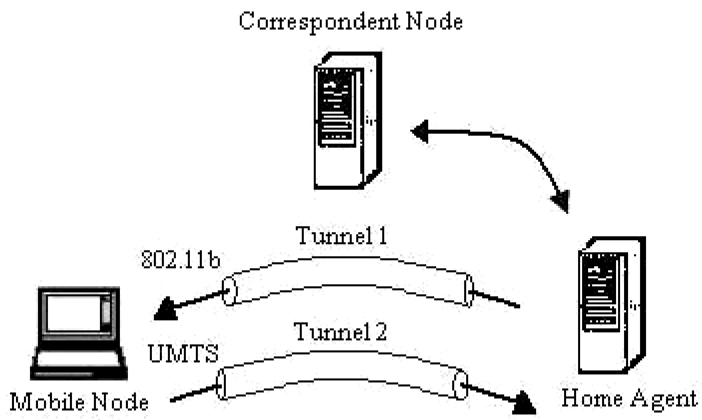


**Figure 11: The MN/HA captures all packets, regarding tunnel.**

The handover is complete when both the MN and the HA have redirected the traffic to the tunnel associated with the handover target's network. This is illustrated in fig.12. The time it takes for SEMO to carry out the entire handover procedure is tied to the handover target network's round trip time (RTT), see fig.19 section 5.1.

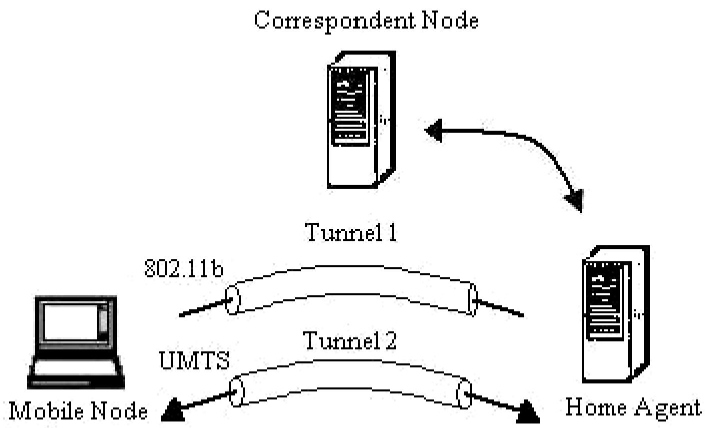


**Figure 12: The handover is complete.**

## *4.3 Policy model*

In the existing Internet infrastructure, there exists no support for a network server, as a NCHO/ MAHO system demands. This argument strongly speaks for designing SEMO as a MCHO system. Another advantage with a MCHO system is that the MN does not need to send any context parameters to a network server, as this would be expensive and capacity demanding in a scenario when for example UMTS is utilized. For these reasons, SEMO is designed as a MCHO system. No policy model was implemented in the previous prototype.

SEMO follows the IETF proposed policy model [26]. The Policy Repository (PR), Policy Decision Point (PDP) and the Policy Enforcement Point (PEP) are located in the MN. The PDP is the method that continuously evaluates and takes policy decisions regarding which network interface that is the most suitable network to utilize. When the PDP decides that a handover is motivated, the PEP enforces the decision. The PEP is the method that via routing rules redirects the traffic over the tunnel associated with the handover target network. The PR is divided into several methods and continuously provides the PDP with static and dynamic parameters.

## *4.4 Policy decision*

### 4.4.1 User Interaction

SEMO's goal is to automatically select and utilize the currently most suitable access network. SEMO uses a user-based cost function to evaluate available networks. The cost function's value is the weighted sum of three different network parameters: utilization cost, power consumption and capacity. SEMO is initially configured to set equal weight to the three different parameters. This results in that the parameters are initially equally prioritized. The user can reconfigure the priority of the parameters via SEMO's graphical user interface (GUI), see fig.13. A reconfiguration affects the result of SEMO's cost function in real time. For example, a user may indicate that the cost for the access network is important, and that the battery consumption is not. When the cost function evaluates network, this configuration will affect the cost function to prioritize cheap networks, whereas their power consumption becomes less important.

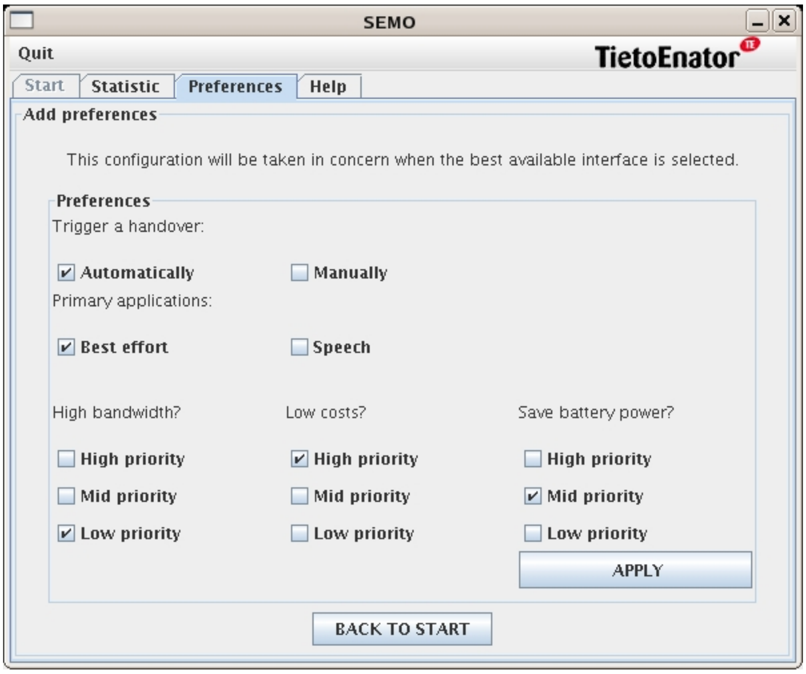


**Figure 13: GUI for user preferences.**

VoIP applications are sensitive to high RTT [45]. A user shall indicate if he/she is going to use a voice over IP application, see the speech box in the upper region of fig.13. When the speech box is enabled, SEMO will avoid handovers to networks with high RTT, unless no other access network is available.

### 4.4.2 Cost function

Cost function $S_i$, see function 1 section 2.3.1, and $f_i$ [27], see function 3 below, are two typical types of cost functions. The coefficient $B_i$ in $f_i$ is interface *i*'s bandwidth, $P_i$ is its power consumption and $C_i$ is its cost.

$$f_i = w_b \ln \frac{1}{B_i} + w_p \ln P_i + w_c \ln C_i \qquad (3)$$

Both cost functions, $S_i$ and $f_i$, calculate a policy value for an access network. The calculated value is the sum of *x* numbers of weighted QoS parameters. The parameters can be cost, capacity, security and/or battery consumptions. $S_i$ represents a cost function that calculates a value of *absolute scale*, meaning that the value itself indicates something. The function $f_i$ represents a function of *relative scale*, meaning that the calculated value itself does not indicate anything specific. The value first becomes interpretable when it is compared to another calculated value. Functions of relative scale often use the natural logarithm. The main differences between the two cost functions are:

- $S_i$ calculates a value of absolute scale, whereas $f_i$ calculates a value of relative scale.

- $S_i$ requires coefficients that must be obtained from a well-tuned function. In other words, the coefficients must be determined through testing. In $f_i$, every parameter is calculated in the same way without any undefined coefficients. If a high parameter value has a positive effect, e.g. bandwidth, that parameter will contribute with the value *ln (1/X)*. See parameter one in $f_i$. If a high parameter value has a negative effect, e.g. cost/hour, that parameter will contribute with the value *ln (X)*. See parameter three in $f_i$. This calculation rule makes $f_i$ easy to extend with new parameters. $S_i$ is not so easy to extend, due to the fact that a new parameter needs a new undefined coefficient.

- If a network interface offers twice the bandwidth, but twice the cost as another network interface, $f_i$ will consider these two networks equally good (if both parameters are equally weighted). For example, if a network interface offers 2 Mbit/s to the cost of 40 cents/hours, that network interface will be equally valued by an interface that offers 1 Mbit/s to the cost of 20 cents/hour. Unlike $f_i$, $S_i$ will not consider the two mentioned network interfaces as equally good.

- $f_i$ and all other cost functions that use the natural logarithm is not defined when QoS parameters are zero. This is because the natural logarithm is undefined for zero. This means that $f_i$ can not handle a scenario where the cost for a network interface is zero. This is a disadvantage, when scenarios exist where e.g. a WLAN can be utilized without charge. $S_i$ is defined for all parameter values.

SEMO is using $f_i$ as cost function. The motivation for SEMO to use $f_i$ as cost function is its characteristic of considering an interface that offers twice the bandwidth, but twice the cost as another equally good. This behaviour makes a cost function of relative scale more general and easier to use in an unspecified environment, than a function of absolute scale. This is an advantage, because parameters for access networks tend to be all but specified. A well-tuned cost function of absolute scale might however be considered as a better choice in a specific environment where all network parameters such as cost, battery consumption, capacity are pre-known.

SEMO's cost function uses the natural logarithm and is therefore undefined for parameters with the value of zero. In SEMO's case, the only parameter that can be zero is the cost parameter. SEMO will therefore add the following rules that are applied when a network interface is free of charge:

- If all available access networks are free of charge, the cost parameter will be excluded, and their cost function values based on the power consumption and capacity parameter will be used to separate the networks.

- If an access network is free of charge and another network is not, the cost of the "free network" is assumed to be 1 cent/hour. This assumption will not indicate that the network is free of charge, but the network is nevertheless considered as extremely cheap.

All written papers concerning cost functions in policy-based vertical handover system have focused on the actual design of the cost function. None has reflected over how to obtain the cost function's parameters in the existing Internet infrastructure. This is a problem, because not even the most intelligent cost function can fairly evaluate access network if its parameters are incorrect. SEMO will try to estimate the parameters with an over time perspective. If time is not taken into perspective, the results from the cost function can only be seen as a snap-shot of which network that should be utilized. For example, assume that a user gives highest priority to long lasting batteries. Further assume that there exist two network interfaces: one power consuming interface with high bandwidth, and one interface with low bandwidth and low power consumption. The naive decision maker would appoint the second network as the most suitable, but this is not always the correct choice. An interface with high bandwidth may reduce transmission times, and with that, reduce the overall power consumption. SEMO will introduce this long term perspective when a network's power consumption, cost and capacity is estimated with the goal to accurately evaluate available access networks.

### 4.4.3 Power consumption parameter

To be able to utilize a certain access network, an MN has to have a network interface card (NIC) associated with it. NICs consumes more battery power in active mode, e.g. during a file transmission, than in idle mode. Table 3 presents the power consumption for an 802.11b [46] and a UMTS network card [47]. These values are used throughout calculations examples in this section. Worth mentioning is that the indicated *active* mode parameter is the maximum power consumption for the two cards. Furthermore, power consumption values for NICs from one manufacture differ from another, so the values in table 3 are not general. The MN's internal architecture and device drivers also affect the total power consumption associated with a certain network interface [48]. This factor has been excluded from the table.

**Table 3: Power consumption for an 802.11b and a UMTS NIC.**

| | Idle | Active | Capacity |
|---|---|---|---|
| 802.11b | 100 mA | 370 mA | 11 Mbps |
| UMTS | 190 mA | 800 mA | 300/64 kbps |

For the power consumption associated with a certain network, it exist a relation between the network's bandwidth capacity and time its NIC spend in either transmission or idle mode. For example, if a certain amount of data is sent each second, a NIC associated with a network with high capacity will spend less time in transmission mode than a NIC associated with a network with low bandwidth. If the values from table 3 are used when the power consumption for the 802.11b and the UMTS are compared, the power consumption and traffic load relation leads to the diagram as fig.14 illustrates. As seen, due to higher capacity, the power consumption for the NIC associated with the 802.11b interface does not increase as much as the NIC associated with the UMTS interface when the traffic load increases.

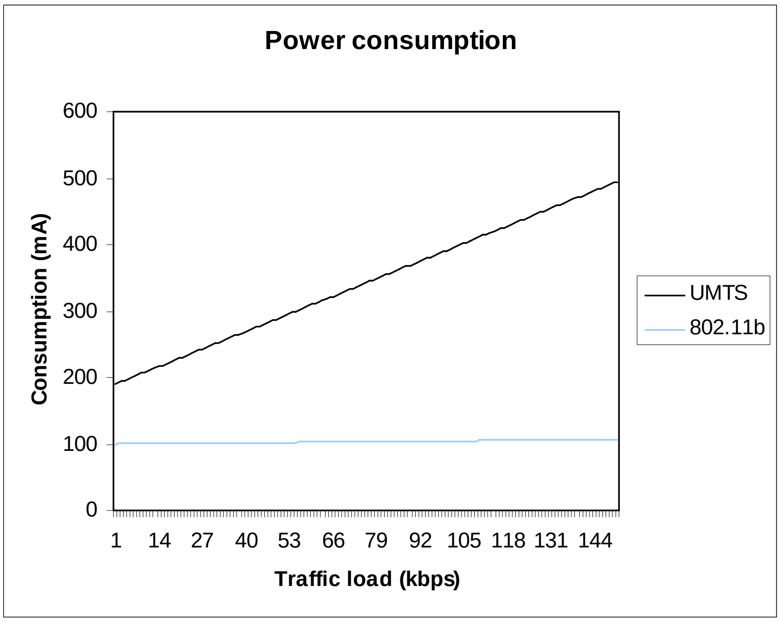


**Figure 14: Calculated traffic load and power consumption relation based on values from table 3.**

When traffic rates are lower than the maximum capacity for every available access network, SEMO will use function 4 to calculate their power consumption. SEMO is able to discover low traffic load by monitoring the MN's traffic. In function 4, $r$ is the current traffic load, $c$ is the access network's capacity and $P$ is the active and idle power consumption associated with the network.

$$P_i = \frac{r}{c} * P_{active} + (1 - \frac{r}{c}) * P_{idle} \qquad (4)$$

Fig.15 illustrates traffic measurements during an instant messaging chat. Due to the constant low traffic load, this is a scenario were SEMO will utilize function 4 to calculate a network's power consumption.

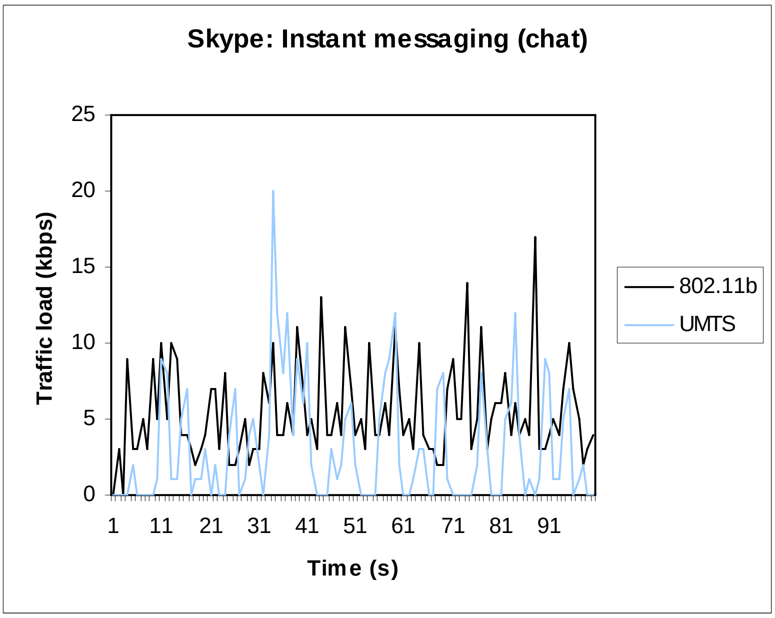


**Figure 15: Traffic generated by a chat session.**

Function 4 can nevertheless only be applied if the traffic load is lower than the maximum bandwidth capacity for every available access network. This is because function 4 only gives a snap-shot value of network interfaces' power consumption. In downloads scenarios, SEMO will use function 5 instead. This is because in a scenario, when for example a large Internet homepage shall be downloaded, a network with high capacity will in the nearest moments spent less time in active transmission mode than an interface with lower capacity. Fig.16 illustrates such a scenario where two large Internet homepages, both slightly smaller than 1 MByte, were downloaded via an 802.11b and a UMTS network.

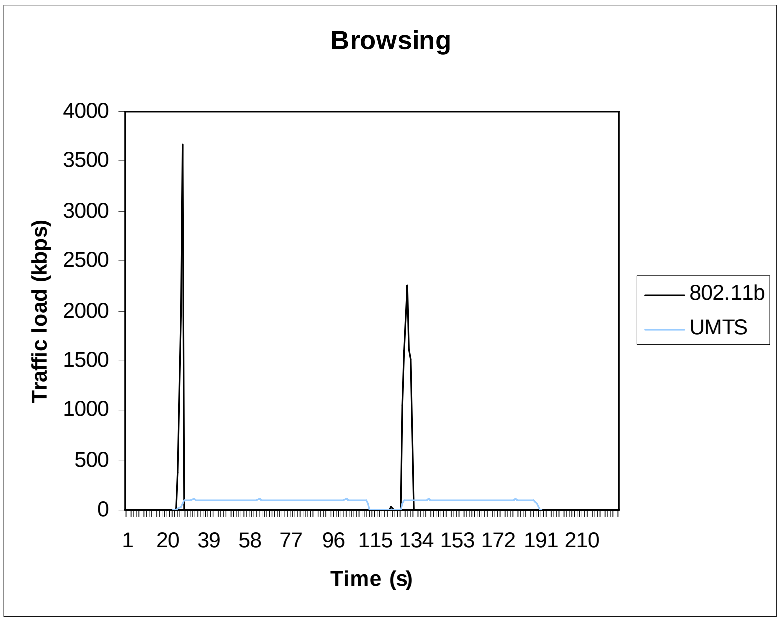


**Figure 16: Traffic generated by browsing.**

Fig.16 shows that the 802.11b network, due to higher bandwidth, spend less time in transmission mode than the UMTS network. SEMO discovers downloads by registering traffic rates near the maximum capacity of the currently utilized access network. As mentioned, when traffic rates are high, SEMO will use function 5 to estimate the network's power consumption.

$$P_i = \frac{a}{c_i} * P_{active} + (\frac{a}{c_{\min}} - \frac{a}{c_i}) * P_{idle} \qquad (5)$$

In function 5, $a$ is the reaming amount of data to download, $c_i$ is the current interface capacity, $c_{min}$ is the capacity of the slowest interface and $P$ is the interface's active and idle power consumption. SEMO will estimate $a$ by predicting the future average download size based on the user's former download sizes. If former download were small, SEMO assumes that future downloads also will be small. If a download is larger than predicted, SEMO will extend the predicted download size with half its original size. A download is registered when chunks of high traffic rates near the interface maximum capacity is detected. The download size is the total amount of data that the chunk consists of. During the work it was experienced that the download traffic flows never lose its rate in download scenarios, at least not long enough to separate "high traffic" chunks.

### 4.4.4 Cost parameter

Network providers usually charges network usage in these three units/ways:

- By the amount of transferred data. UMTS subscribers are usually charged per bytes sent.

- Per usage time. WLAN in e.g. Internet cafes or centre of cities, often charges a user per hour or minute.

- For example, a mobile phone subscriber can have certain free amount of data per month and traffic that exceeds the free amount is charged per sent bytes.

The fact that networks often use different units makes it difficult to compare the cost for different networks. How much network traffic a user generates can be used to calculate the actual cost for utilizing a network over a period of time. If a user generates large amounts of traffic, a network that charges a user per connection time is generally more likely to be cheaper than a network that charges a user per sent data. One problem is however that it is often impossible to predict the future network load. SEMO uses a prediction model that is based on the following assumption: If the recent network traffic was high, it is likely that it will continue to be high in the nearest future. If the recent network traffic was low, it is likely that it will continue to be low in the nearest future. SEMO will continuously keep a ten minute history window of the user's network load. The window will be used to estimate the next ten minutes' traffic rate. A relatively short history window has deliberately been chosen so changes in the network load will swiftly affect SEMO's cost predictions.

Two example scenarios are presented below, where networks are charged with different units. In the scenarios, a suggestion is presented how the cost for the different networks can be compared. It is assumed in both examples that the total generated network traffic for the latest ten minutes is 6 Mbytes.

**Example 1**:

- *Network one*: Charges the user after how long effective time the user has utilized the network. The cost of the network is 15 cents/min.

- *Network two*: Charges the user per bytes sent. The cost of the network is 20 cents/ Mbytes.

The cost unit of *Network two* is first converted to its estimated cents/minute value. When the two networks have the same cost unit, their cost can easily be compared. The estimated cost for *Network two* in cents/min is:

- 20 cents/Mbyte * Assumed network traffic/min = 20 * 6 Mbyte/10 = 12 cents/min.

According to the calculation, the cost of *network two* is 12 cents/min. By doing this conversion, SEMO can appoint *network two* as cheaper than *network one*.

**Example 2:**

- *Network one*: In the centre of a city, a user is offered network connectivity via an 802.11b network. For network usage, a user is charged for a fixed period of time, 3 dollars/30 min.

- *Network two*: Charges the user per bytes sent. The cost of the network is 20 cents/Mbytes.

The basic idea is to predict if it is cheaper to rent *network one* or if it is cheaper to only use *network two*. If the 30 minutes fee for *network one* is paid, the cost for using *network two* increases indirectly. If *network two* is utilized even if the fee is paid, the fee can be seen as wasted. When comparing the cost for *network one* and *network two*, it is assumed that the user will stay in the centre of the city for a longer period of time. If the user is just passing through the city, it is generally cheaper to not pay the half hour fee.

Initially, it has to be decided if *network one* is more likely to be cheaper than *network two* for the next 30 minutes:

- Cost of *Network one* the nearest half hour: 3 dollars.

- Estimated cost of *Network two* the nearest half hour: 20 cent/Mbyte * assumed network traffic/min * 30 min = 20 * ((6/10)*30) = 3.6 dollars.

Under these conditions, *network one* is estimated to be cheaper than *network two* for the next 30 minutes. If *network one* is used, the fixed price of the 3 dollars fee must be paid, even if the network is only used for a shorter time than 30 minutes. If the fee is paid, the cost for network one can be seen as zero for the rest of the paid 30 minutes. As a chain reaction of the paid fee, the cost of *network two* will be indirectly affected during this time. The cost of *network two* can be converted to cost/time, e.g. cost/seconds and is during this time the sum of the ordinary cost of the network plus the cost for *network one* during the same period of time. In other words, if the fee for *network one* is paid, the cost for using *network two* one second can be calculated as the sum of its own cost/second plus the cost for using *network one* for one second.

In a scenario when a subscriber has a certain amount of free data in combination with a fixed price for cost/sent bytes, the network can be considered as free until the free amount of data is ended. If the network has very low capacity, and the traffic is high, it might still be wise to utilize a network that is charged per usage time. It is because it is probably cheaper over time to use the free amount data were the traffic is low, so the free data is not wasted within a short period of time.

### 4.4.5 Capacity parameter

Different network interfaces offers different capacity, see section 2.1.6. When a network should support high QoS, bandwidth is usually considered to be the most important parameter [20]. Unfortunately, it is very difficult to measure an access network's available bandwidth [49]. To estimate a network's capacity, SEMO will calculate the network's Relative Network Load (RNL) [50]. RNL is a mathematical formula that calculates a quality value for a network, based on RTT and jitter values, with the goal to estimate its capacity. Fig.17 shows the RNL formula.

$$B_i = RNL_n = \bar{x} + V_n$$

*where*

$$\bar{x}_n = \frac{1}{h} x_n + \frac{h-1}{h} \bar{x}_{n-1}$$

$$V_n = \frac{1}{h}(x_n - \bar{x}_n)^2 + \frac{h-1}{h} * V_{n-1}$$

**Figure 17: The RNL formula.**

Function $\bar{x}_n$ calculates the mean value of the RTT for packets sent between the MN and the home agent. Function $V_n$ calculates the variance of the measured RTT's. In both function $\bar{x}_n$ and function $V_n$, the variable $h$ determines the size of the history window for the mean values. For example, $h = 5$ will result in that the most recent value will contribute with 20 per cent to the total $\bar{x}_n / V_n$ mean value.

# Chapter 5

# Architecture

## 5.1 System overview

As mentioned, SEMO uses M-MIP to support seamless mobility and consists of an MN and an HA application. The HA captures all incoming packets destined to the MN's home address and tunnels them to the MN's current address. The HA is able to capture the packets by answering Address Resolution Protocol (ARP) requests on behalf of the MN's home address.

By using SEMO's MN application, an MN can automatically discover and establish a connection to a LAN, a WLAN and/or a UMTS network. When a connection is established, a tunnel associated with the new access network is set up between the MN and the HA. Based on a user-based policy decision, the MN appoints the currently most suitable access network and sends traffic over the tunnel associated with the appointed interface. The tunnels are implemented with help from the open source program *openVPN* [51]. A handover is performed by redirection of traffic from the tunnel associated with the current network to the tunnel associated with the handover target network. The redirection is performed via changes in routing rules. To be able to assure network connectivity, synchronic handovers and performing capacity measurements, registration messages are periodically sent over all available networks. Fig.18 illustrates a system overview in a scenario where the MN is connected to both an 802.11b and a UMTS network.

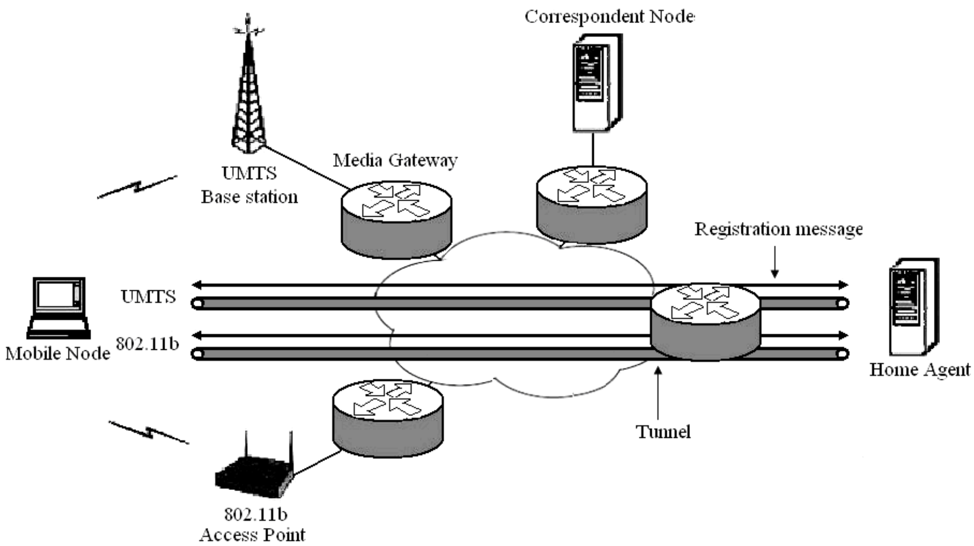


**Figure 18: System overview when the MN is connected to both an 802.11b and a UMTS network.**

Fig.19 illustrates the registration messages exchanged in SEMO's initiating and handover procedure, when an 802.11b and a UMTS network are available. Initially, both the 802.11b and UMTS interface sends registration messages to the HA, receive a reply and setup their tunnels. The 802.11b interface is first to register, so traffic is sent over its tunnel. After the initiation procedure, registrations messages will be periodically sent over both interfaces. This message exchange scenario will be kept, as long as no handover decision is made. When the MN makes a handover decision, a registration message is sent over the UMTS network to inform the HA to redirect the traffic to its associated tunnel. The HA informs the MN that the traffic is redirected via a reply.

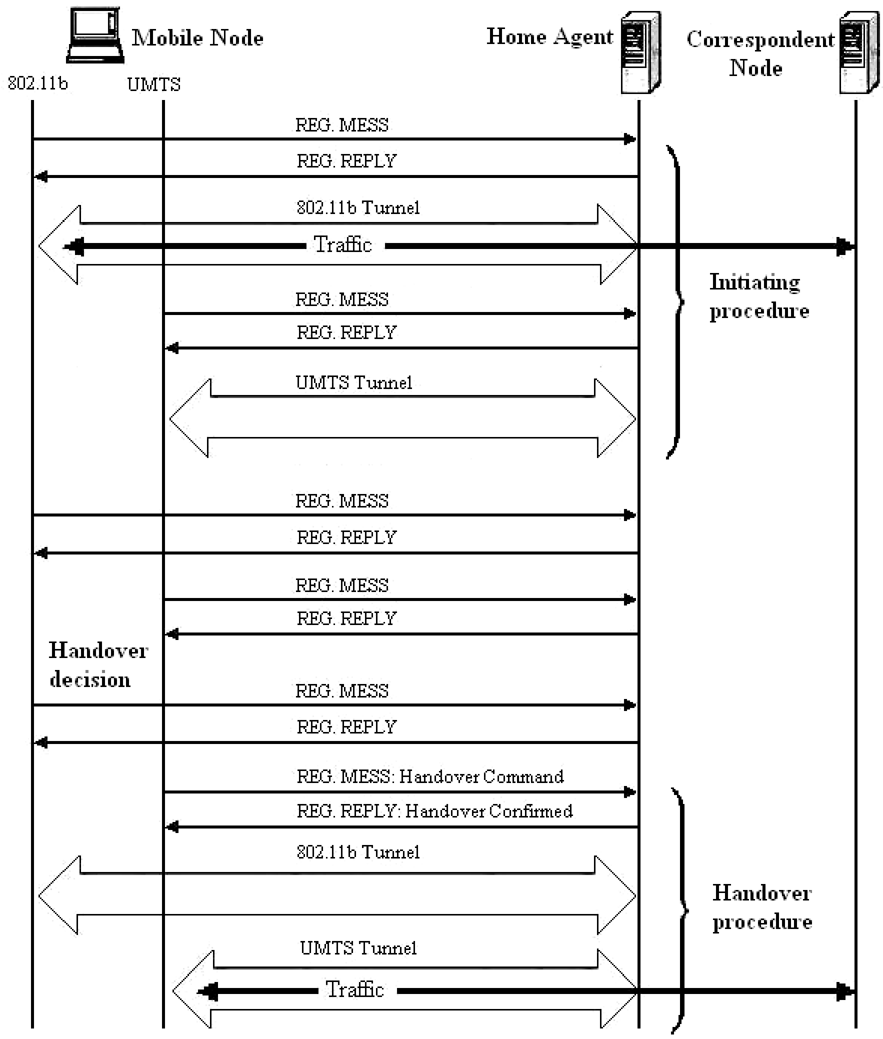


**Figure 19: SEMO's registration message exchange.**

## *5.2 Architecture of Mobile Node application*

Fig.20 shows a class diagram for the central classes of the MN application. The application is started via the *GUI* class. All network interfaces inherit from the base class *Interface*. New types of network interfaces are easily extended, due to that only three interface specific methods have to be implemented. Every interface run on a separate thread and periodically sends registration messages to the HA application. The *PolicyEngine* class has access to all interfaces through the *vector ifaceVec*. The *PolicyEngine* continuously evaluates the interfaces and decides which interface that is most suitable to utilize. The PDP is located in the *PolicyEngine* class. A handover is performed by the interface class, so the PEP is located in the interface class. The *Traffic* class provides the *PolicyEngine* with network traffic data and the *UserProfile* class keeps track of the traffic history patterns. These classes together with the *interface* class assemble the PR.

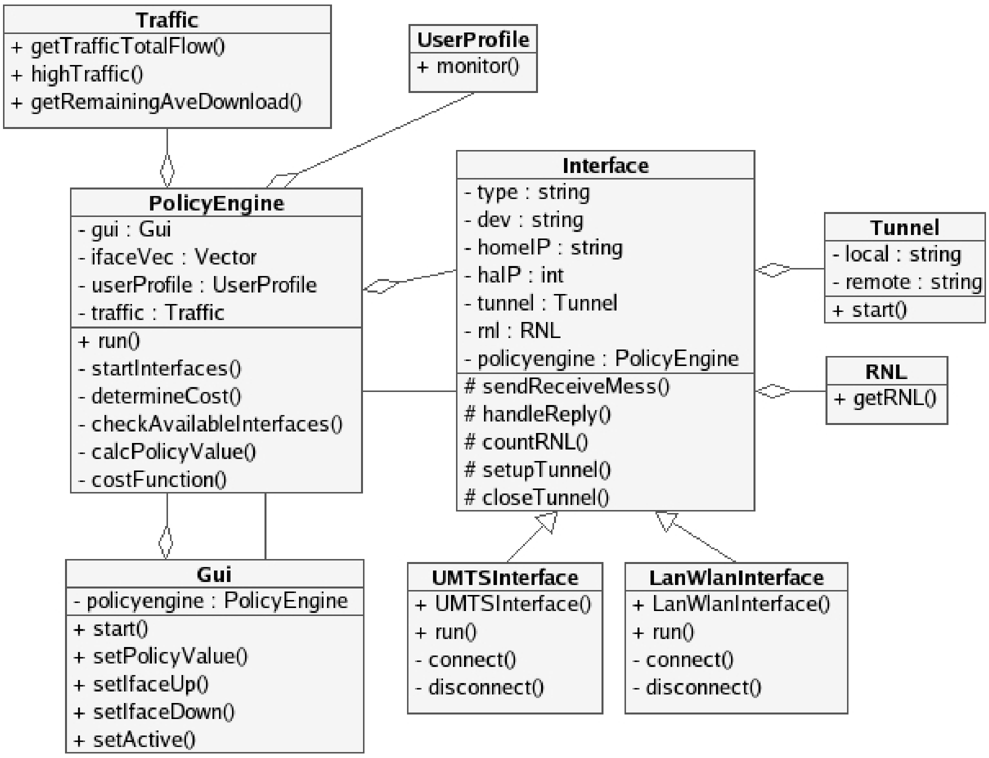


**Figure 20: Class diagram for SEMO's Mobile Node application.**

### 5.2.1 Algorithms

When an interface thread is started, its *run* method is launched. The run method use the algorithm illustrated in fig.21 to manage network connectivity aspects such as establish a connection, set up the tunnel associated with the interface, send and receive registration messages from the HA application etc.

1. While thread is running
   - 1.1 While coverage not found
     - 1.1.2 Seek for nearby networks
   - 1.2 Setup tunnel associated with the found network
   - 1.3 Add routing rules
   - 1.4 While connectivity exist
     - 1.4.1 Send/Receive register message
   - 1.5 Alert the PolicyEngine of lost connectivity
   - 1.6 Remove tunnel

**Figure 21: Algorithm that manages connectivity aspects for an interface.**

SEMO's PDP is located in the PolicyEngine class and the algorithm illustrated in fig.22 is used to continuously evaluate network interfaces. If the currently utilized network is not appointed as the most suitable network, a handover is performed.

1. While tread is running
   - 1.1 Check interfaces' connectivity status
   - 1.2 Check traffic load
   - 1.3 For every interface with valid connection status
     - 1.3.1 Calculate the interface's cost, power consumption and capacity parameter
     - 1.3.2 Calculate their policy value
   - 1.4 Add a hysteresis margin to the currently used interface's policy value
   - 1.5 If the currently used interface's policy value is not the "best"
     - 1.5.1 Handover to the interface with the best policy value
   - 1.6 Sleep

**Figure 22: Algorithm for the PDP.**

## 5.3 Architecture of Home Agent application

Fig.23 shows a class diagram for the central classes of the HA application. The *RegRecv* handles all incoming registration from an MN and delivers the registrations to an instance of the class *Agent*. The class *MobileNode* represents an MN. An *Agent* has a *vector* with *MobileNode* as entries, where all registered MNs are gathered. A *MobileNode* has an instance of the *MyTimer* class. A *MobileNode's* timer is restarted every time the HA application receives a registration message associated with that *MobileNode*. If the timer expires, its associated *MobileNode* entity will be removed.

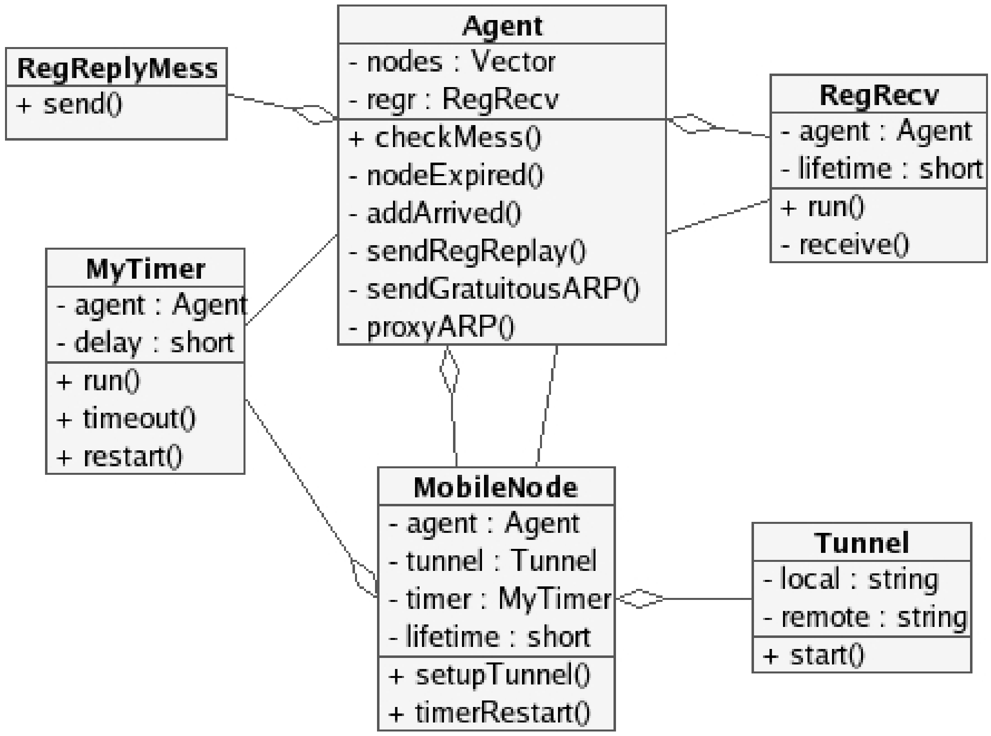


**Figure 23: Class diagram for SEMO's Home Agent application.**

### 5.2.1 Algorithm

The HA application's main purpose is to receive registration messages from an MN, handle the registrations properly and respond the MN with a reply message. The method *checkMess*, located in the class *Agent*, handles this. An algorithm for *checkMess* is illustrated in fig.24. The method takes a received registration message as argument.

1. If the registration came form a unregistered MN
   - 1.1 Register that the MN exists.
   - 1.2 Setup a tunnel associated with the MN
2. Else
   - 2.1 Restart the timer associated with the MN
3. If a handover is requested
   - 3.1 Send traffic over the tunnel associated with the MN
4. Send a reply message to the MN
5. Send out an ARP reply/request associated with the MN's home address

**Figure 24: Algorithm for the *checkMess* method.**

## *5.4 System requirements*

### 5.4.1 Mobile Node application

The system requirements for the MN application are:

- Linux Platform. The MN application has mainly been tested on a laptop running the Linux distribution *Fedora core 4* [52], with kernel 2.6.11.
- Java™ 2 Runtime Environment, Standard Edition v 1.5.0 (J2SE).

The Linux kernel has to be compiled to support network devices:

- Universal TUN/TAP device driver
- PPP (point-to-point protocol)
- IP: advanced router
- IP: policy routing

The open source packages *openVPN* and *iproute* must be installed. The MN must have a public home IP address at the same local network as the HA. The MN's current IP address must also be public.

### 5.4.2 Home Agent application

The system requirements for the HA application are:

- Linux Platform. The HA application has mainly been tested on a laptop running the Linux distribution *Fedora core 4*, with kernel 2.6.11.
- Java™ 2 Runtime Environment, Standard Edition v 1.4.2 (J2SE).

The Linux kernel has to be compiled to support Universal TUN/TAP device driver and *openVPN* has to be installed. The IP address that the HA uses must be public.

# Chapter 6

# Results

## 6.1 Goal fulfillment

All requirements for the prototype of this thesis have been fulfilled. The requirements are presented in section 1.5.1 and table 4 shortly describes how the different requirements have been fulfilled.

**Table 4: Requirements realization description**

| Requirement | Fulfilled | Description |
|---|---|---|
| Req.1 | Yes | For selecting available network interfaces, SEMO uses a user-based policy with cost, power consumption and capacity as parameters, see section 4.4.1. |
| Req.2 | Yes | When SEMO loses connection to an access network, reestablishments attempts is performed automatically. A establish connection can be verified by the Linux command *ifconfig*. |
| Req.3 | Yes | A handover is automatically performed when the currently utilized access network loses connection or the policy triggers a handover. The network analyzer tool *Ethereal* [53] can be used to monitor and verify the handover. |
| Req.4 | Yes | A handover can be triggered manually via SEMO's GUI, see *User's Guide* appendix A. |
| Req.5 | Yes | SEMO automatically discovers and establish connections to a WLAN, a UMTS or a LAN network. |
| Req.6 | Yes | Measurements regarding instant throughput during a hard and a soft handover have been performed, see section 6.2. |
| Req.7 | Yes | The implementation is divided into modules and design so new network technologies are easily extended, see section 5.2. |
| Req.8 | Yes | SEMO requires Linux as platform, see section 5.4. |
| Req.9 | Yes | The MN and the HA application are implemented in Java. The HA uses short C programs form sending ARP requests. Both applications use shell scripts. |
| Req.10 | Yes | The report is written in English and includes an analysis, design and user's guide. |

## 6.2 Handover measurements

SEMO simultaneously utilizes both the current and the handover target network during a handover procedure. SEMO achieves this by establishing multiple tunnels, one for each network. By this design, SEMO can perform a soft handover. The previous prototype utilized a single tunnel, and is therefore only able to be utilizing one interface at the time. This design leads to hard handovers. This section presents throughput measurements during a handover for the two prototypes. The handover is manually triggered from an Ethernet (100 Mbps) to an 802.11b (11 Mbps) network. Fig.25 illustrates the testbed that were used during the measurements.

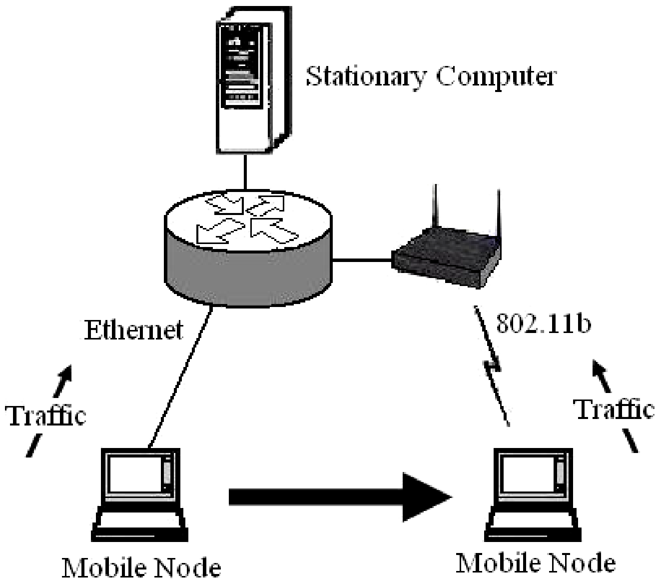


**Figure 25: Testbed used in vertical handover experiments**

To perform a test under high traffic load, *the Distributed Internet Traffic Generator* (D-ITG) [54] was used to generate traffic from the MN to a stationary computer. A D-ITG receiver server was started at the stationary computer and a D-ITG client was started at the MN. The client continuously sent 2500 UDP packets/second with the packet size of 567 bytes to the server. This resulted in a network load of 11.52 Mbps, which is higher then the 802.11b maximum capacity (theoretical 11Mbps, practical 5.5 Mbps). Such a high network load was chosen to test the hard and soft handover mechanism under a high traffic load. The throughput from the MN to the stationary computer was measured in both handover tests.

### 6.2.1 Hard handover from Ethernet to 802.11b

As mentioned above, the maximum traffic load is set to 11.52 Mbps. When the MN utilizes the Ethernet connection, the throughput reaches the maximum rate, see fig.29. The handover is triggered just before eight and a half second. During the handover, the tunnel, that was initially bound to the Ethernet network is broken and rebound to the 802.11b network. Fig.26 illustrates the throughput measurement.

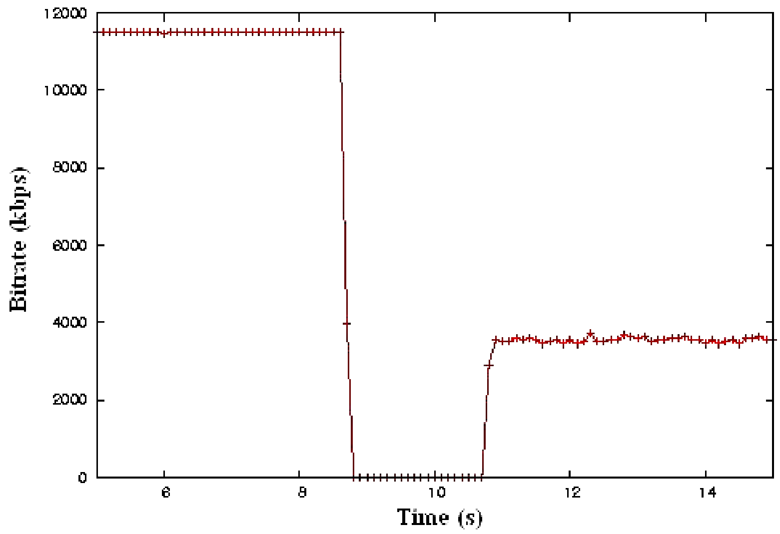


**Figure 26: Throughout results for a UDP flow during a hard handover from Ethernet to 802.11b**

Due to the handover mechanism design, a connectivity break occurs during a handover. In this experiment, the throughput was halted in 2.07 seconds. Furthermore, 2500 packets were sent every second. This leads to the conclusion that approximately 5175 packets failed to be delivered.

### 6.2.2 Soft handover from Ethernet to 802.11b

As in the earlier handover experiment, when the tunnel associated with the Ethernet network is utilized, the throughput reaches the maximum 11.52 Mbps. The handover is manually triggered just before ten seconds, see fig.27.

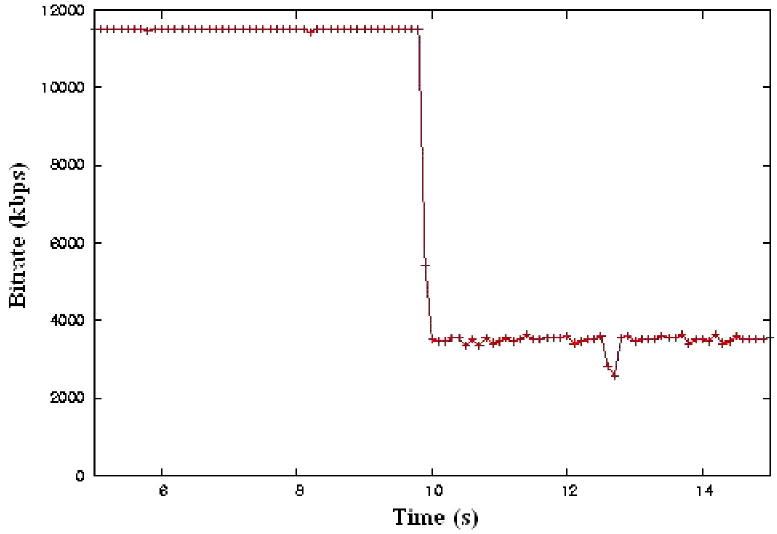


**Figure 27: Throughout results for a UDP flow during a soft handover from Ethernet to 802.11b**

Due to SEMO's handover mechanism, the tunnel associated with the 802.11b network connection is already established before the handover. As seen in fig.30, the handover did not result in any throughput interruption. In this experiment, all sent packets were delivered.

## *6.3 Moving-out and Moving-in measurements*

This section presents policy value measurements when SEMO leaves and enters an 802.11b access network. The purpose with this section is to show how SEMO behaves in such scenarios. The measurements were conducted inside a public building that had an 802.11b network that covered the centre of the building and a UMTS network that covered the entire building, see fig.28. The radius of the 802.11b coverage was approximately 50 meters.

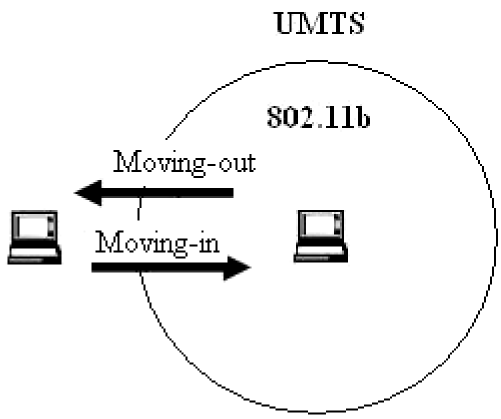


**Figure 28: The moving-out and moving-in scenario.**

The cost, capacity and the power consumption parameter were equally prioritized, so SEMO used function 6 to calculate the policy value, PV, for interface *i*:

$$PV_i = \frac{1}{3}\ln Capacity_i + \frac{1}{3}\ln Cost_i + \frac{1}{3}\ln Power\ consumption_i \qquad (6)$$

Table 5 presents the cost and the power consumption values used throughout the measurements.

**Table 5: The parameters used in the moving-in/moving-out measurement.**

| Interface | Cost | Battery (Active/Idle) |
|---|---|---|
| 802.11b | 300 cents/hour | 370/100 mA |
| UMTS | 30 cents/Mbyte | 800/190 mA |

As table 5 shows, the two networks used different cost units. When SEMO had access to both the 802.11b and the UMTS network, SEMO converted the cost of the UMTS to its predicted cost/hour value, see example 1 section 4.4.4. The traffic load was approximately 30 Kbps equally divided between the up and down link of the currently utilized network interface. SEMO classified this traffic rate as low and used function 4, see section 4.4.3, to calculate the power consumption associated with the interfaces.

When SEMO compared the two networks, the mean values of their three latest policy values were used. This approach was used to increase stability, supplementary to the hysteresis margin, and thereby avoid handover ping-pong effects.

### 6.3.1 Moving-out from the 802.11b network

In this scenario, the MN started at the 802.11b AP and moved away from the AP with a velocity of approximately 1.5 m/s. The 802.11b network was initially the currently utilized network, but due to lesser coverage, the RNL value for the 802.11b network successively increased and a handover was performed to the UMTS network. The instant policy values from the UMTS and the 802.11b network are illustrated in fig.29.

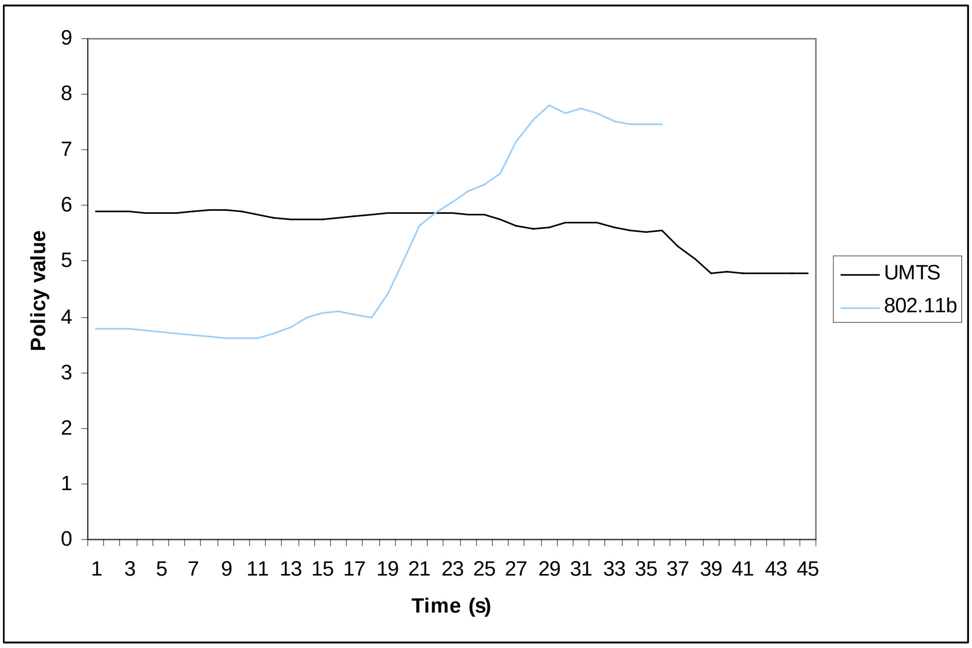


**Figure 29: Instant policy values for the UMTS and the 802.11b networks in the moving-out scenario.**

**Comments:** In this moving-out scenario, the only thing that significantly affected the 802.11b policy value were changes in the 802.11b network's RNL value. SEMO lost the connection to the 802.11b network after approximately 50 meters.

### 6.3.2 Moving-in to the 802.11b network

In this scenario, the MN initially only had UMTS coverage and started to approach the 802.11b AP with a velocity of approximately 1.5m/s. Fig.30 shows the instant policy values for the UMTS and the 802.11b network.

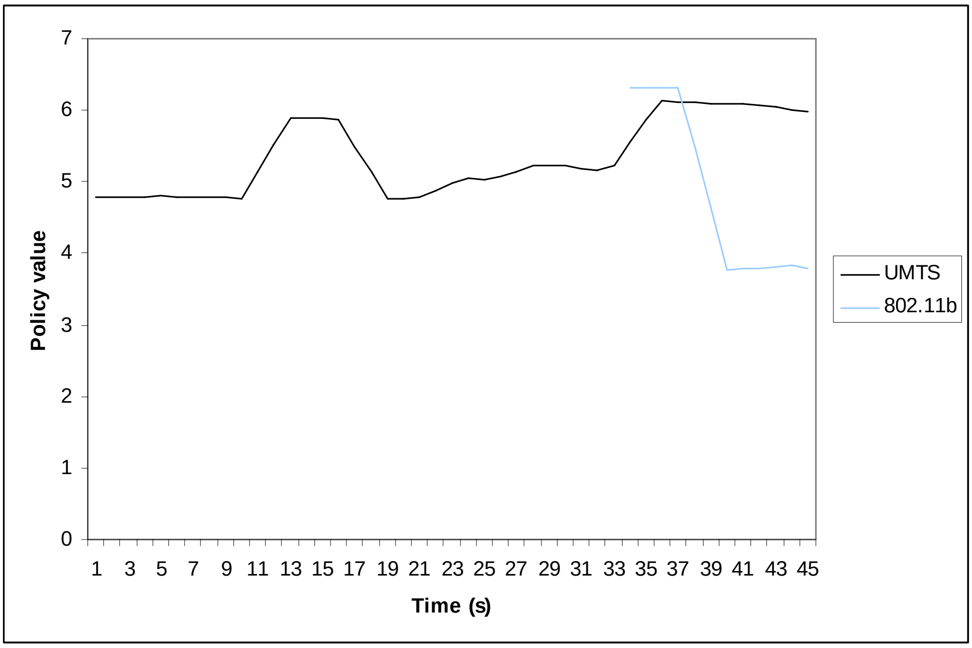


**Figure 30: Instant policy values for the UMTS and the 802.11b networks in the moving-in scenario.**

**Comments:** The RNL value for the newly encountered 802.11b network is initially set to a poor value to prevent a handover based on an overestimation of its capacity. The RNL value for the 802.11b network was far better than the UMTS network's, so SEMO made a policy triggered handover to the 802.11b network. The increase of the UMTS's policy value from 12-17 seconds was due to an increasing RNL value. This was probably caused by the changes in conditions when the MN moved around in the building. In this measurement, the MN established the 802.11b connection approximately 20 meters from the AP. During the work, it was often experienced that SEMO needed to be close an 802.11b AP to establish a connection, much closer than the distance that SEMO could keep a connection alive.

# Chapter 7

## 7.1 Conclusions

This report has presented the seamless mobility prototype SEMO. SEMO is a policy-based prototype that automatically connects to and utilizes the most suitable available access network. The most suitable access network is appointed through a user-based policy decision. A user can affect the policy decision by expressing his/hers preferences regarding the significance of a network's capacity, cost and power consumption. SEMO has introduced a long term perspective when the mentioned parameters are estimated. The estimation is done through analysis and predictions based on the user's traffic load history. SEMO is compatible to the existing Internet infrastructure and uses M-MIP to support seamless mobility. SEMO can perform soft handovers and experiments have shown that SEMO can eliminate packet loss tied to the handover procedure. SEMO can currently utilize UMTS, WLAN and Ethernet, but can easily be extended with new network interface technologies such as WiMAX.

By using SEMO's policy model to determine when to perform a handover, lost connections can be more effectively discovered, than for example if a hard coded policy that always chose WLAN before UMTS was used. This is due to that SEMO policy model discovers when a connection is about to be lost via a poor policy value and swiftly handover to another available network.

I believe that we will see an increasing number of seamless mobility products on the market in the nearest future. Some factors that contribute to better conditions for a product expansion are the increasing number of mobile devices such as PDAs and laptops combined with an increasing number of available WLAN hotspots. One disadvantage for seamless mobility products is poor quality when VoIP applications are utilized over UMTS. I think that it is important to develop new VoIP protocols adapted to UMTS. In the closest years, I think that a mobile phone is better equipped to handle a voice communication, whereas a seamless mobility system similar to SEMO is better equipped to support internet browsing, downloads, chat sessions etc.

All requirements for this thesis has been fulfilled and verified by the internal supervisor at TietoEnator.

## 7.2 Limitations and Future Work

SEMO uses the open source application *openVPN* to tunnel the traffic between the MN and the HA. Linux scripts are also used, especially when a connection to a network is established. By implementing a tunnel and establish connections through application program interfaces (APIs), better control, e.g. via return messages, can be achieved. SEMO periodically sends registration messages outside the tunnels to determine the capacity for an access network, and has a pay-off between the rate of sent registration messages and the ability to discover a lost connection. By using the virtual Linux device *TAP*, a tunnel can be implemented and with that, enable registration messages through the tunnels. When regular traffic exists, that traffic could replace the registration messages. By this approach, a faster discovery of lost connections could be achieved, without having to send more registration messages.

Another recommended future work is to make duplication of traffic over different access networks possible, so no packets are lost even if the currently utilized network unpredictably disconnects. This would be extra useful when MN travels in high velocities. Furthermore, a higher QoS could be achieved if different applications/ports use different networks. For example, a VoIP application could use the 802.11b network simultaneously as an ftp-application utilizes the Ethernet connection.

# References


[1] Chen L.-J., Sun T., Chen B., Rajendran V., and Gerla M., A smart decision model for vertical handoff, *In Proceedings 4th ANWIRE International Workshop on Wireless Internet and Reconfigurability*, Athens, Greece, 2004.

[2] http://www.thefreedictionary.com/heterogenous, 2005-12-12.

[3] Encyclopedia of Networking, Mitch Tulloch, 2nd edition, ISBN:0735605734 , 2000.

[4] www.webopedia.com/TERM/H/heterogeneous_network.html, 2005-12-27

[5] M. Ylianttila, Vertical Handoff and Mobility - System Architecture and Transition Analysis, Doctoral thesis, Department of Electrical and Information Engineering, University of Oulu, Finland. ACTA Univ. Oul. C220, 2005.

[6] McNair, J. and F. Zhu, Vertical Handoffs in Fourth-Generation Multinetwork Environments*, IEEE Wireless Communications* June 2004.

[7] M. Katz, F.H.P. Fitzek, On the Definition of the Fourth Generation Wireless Communications Networks: The Challenges Ahead, in *International Workshop on Convergent Technology (IWCT) 2005*, Oulu, Finland, June 2005.

[8] Andrea Calvagna, Giuseppe Di Modica, A user-centric analysis of vertical handovers, *Proceedings of the 2nd ACM international workshop on Wireless mobile applications and services on WLAN hotspots*, Philadelphia, pp. 137–146, 2004

[9] http://www.commsdesign.com/csdmag/sections/cover_story/showArticle.jhtml?-articleID=16500515, 2005-12-12

[10] Mobile Communications, Jochen Schiller, 2nd edition. ISBN: 0321123816, Sep 11 2003

[11] C. Perkins, Ed. IP Mobility Support for IPv4, RFC 3344, Aug. 2002.

[12] David B. Johnson, Charles Perkins, and Jar/ Arkko. Mobility support in 1Pv6. *Internet-Draft draft-ietf-mobileipipv*6 -18.txt, 1ETF Mobile 1P Working Group, INTERNET-DRAFT, June 2002.

[13] D. Johnson and C. Perkins, Route Optimization in Mobile IP, Internet Draft draft-ietf-mobileip-optim-11.txt (work in progress), September 2001.

[14] Hinden, R. and S. Deering, Internet Protocol Version 6 (IPv6), Addressing Architecture, RFC 3513, April 2003.

[15] S. Thomson, T. Narten, IPv6 Stateless Address Autoconfiguration, RFC 2462, December 1998.

[16] M.Ylianttila, M,Mäkelä J & Pahlavan K, Analysis of handover in a location-aware vertical multi-access network. *Computer Networks 47(2)*:185-201, 2005.

[17] Mark Stemm, Randy H. Katz, Vertical Handoffs in Wireless Overlay Networks, *ACM Mobile Networking* (MONET), 1998.

[18] R. Chandra, P. Bahl, P. Bahl, MultiNet: Connecting to Multiple IEEE 802.11 Networks Using a Single Wireless Card, *Proceedings of IEEE Infocom,* 2004.

[19] Vidales P, Patanapongibul L, Mapp G.E and Hopper A, Experiences with Heterogeneous Wireless Networks - Unveiling the Challenges. *Second International Working Conference on Performance Modelling and Evaluation of Heterogeneous Networks* (HET-NETs04) July 2004.

[20] Nilsson, T. Resource Allocation and Service Differentiation in Wireless Local Area Networks. Licenciate thesis. Department of Computing Science, Umeå University, Sweden, 2005, ISBN-91-7305-907-2

[21] C. Eklund, R. B. Marks, K. L. Stanwood and S. Wang, IEEE802.16: A technical overview of the Wireless MAN Air Interface for broadband wireless access, *IEEE Communications Magazine*, Vol. 40, No. 6, pp.98-107, June 2002.

[22] Carlos T. Calafate, Pietro Manzoni, Manuel P. Malumbres, On the interaction between IEEE 802.11e and routing protocols in Mobile Ad-hoc Networks, *13th Euromicro Conference on Parallel, Distributed and Network-based Processing* (PDP), Lugano, Switzerland, February 9-11 2005.

[23] http://www.webopedia.com/TERM/b/bandwidth.html, 2005-12-12.

[24] G. Breed, Bit Error Rate: Fundamental Concepts and measurement issues, *High Frequency Electronics, Jan 2003*

[25] M.Kamilova, Policy-based Handovers across Intermediary Multimedia Content Providers in the Wireless Internet, Thesis for a degree of Professional Doctor of Engineering, Enschede, the Netherlands October 2004.

[26] R.Yavatkar, D. Pendarakis and R. Guerin, “A Framework for Policy-based Admission Control”, RFC 2753, January 2000.

[27] H. J. Wang, R. H. Katz, and J. Giese, Policy-enabled Handoffs Across Heterogeneous Wireless Networks, *Proceedings of the Second IEEEWorkshop on Mobile Computer Systems and Applications,* pp. 51-60, Feb. 1999.

[28] K. Chan, J. Seligson, D. Durham, S. Gai, K. McCloghrie, S. Herzog, F. Reichmeyer, R. Yavatkar, and A. Smith, COPS Usage for Policy Provisioning, IETF, RFC 3084, March 2001.

[29] F. Zhu, J. McNair, Optimizations for Vertical Handover Decision Algorithms, *Proc. IEEE WCNC ’04, Mar*. 2004.

[30] H. Morikawa, T. Aoyama, Realizing the Ubiquitous Network: the Internet and Beyond, *Telecommunication Systems*, vol. 25, no. 3-4, pp.449-468, March-April 2004.

[31] Jun-Zhao Sun and Jaakko Sauvola, Mobility Management Reconsideration: Hierarchical Model and Flow Control Methodology, *Proceedings of the 14th IEEE*

*International Symposium on Personal, Indoor and Mobile Radio Communications*, Beijing, China, 2003.

[32] K. Murray, R. Mathur, and D. Pesch, Intelligent access and mobility management in heterogeneous wireless networks using policy. In *ACM 1st International Workshop on Information and Communication technologies*, pages 181.186, 2003.

[33] K. Murray, D. Pesch, Intelligent Network Access and Inter-System Handover Control in Heterogeneous Wireless Networks for Smart Space Environments, *The International Symposium on Wireless Communication Systems 2004,* pp.66- 70, 20-22 Sept. 2004.

[34] http://www.telia.se, 2005-08-26.

[35] S. Aust, D. Proetel, N. A. Fikouras, C. Pampu, and C. Gorg, Policy based Mobile IP handover decision (POLIMAND) using generic link layer information. *In 5th IEEE International Conference on Mobile and Wireless Communication Networks* (MWCN 2003), October 2003.

[36] Zonoozi M, Dassanayake P & Faulkner M, Optimum hysteresis level signal averaging time and handover delay. *Proc. 47th IEEE Vehicular Technology Conference*, Phoenix, Arizona, USA, 1: 310-313, 1997

[37] W. T. Chen, Y. Y. Shu, Active Application Oriented Vertical Handoff in Next-Generation Wireless Networks, *The IEEE Wireless Communications and Networking Conference 2005*, New Orleans, LA, USA, March 2005, accepted November 2004.

[38] http://www.netl.doe.gov/coolscience/teacher/lesson-plans/lesson6.pdf, 2005-12-12

[39] Christer Åhlund, Arkady Zaslasky, Extending Global IP Connectivity for Ad Hoc Networks, *Telecommunication Systems*, 24(2), 2003.

[40] C. Åhlund, R. Brännström, A. Zaslavsky, M-MIP: Extended Mobile IP to Maintain Multiple Connections to Overlapping Wireless Access Networks, International Conference on Networking, April 2005.

[41] R. Wakikawa, Multiple Care-of Addresses Registration, draft-wakikawa-mobileip-multiplecoa-04.txt, INTERNET-DRAFT 20 Jun 2005

[42] X. Zhao, C.Castelluccia, M.Baker, Flexible network support for mobility, In *Fourth ACM International Conference on Mobile Computing and Networking (MOBICOM'98)*, 1998.

[43] L-J.Chen, T. Sun, B.Cheung, D.Nguyen, and M.Gerla, Universal Seamless Handover Architecture in Wireless Overlay Networks, Technical Report TR040012, UCLA CSD, 2004.

[44] C.Chiasserini, F.Cuomo, L.Piacentini, M.Rossi, I.Tinirello, F.Vacirca, Architectures and protocols for mobile computing applications: a reconfigurable approach, *Computer Networks* 545-567, 2004.

[45] http://www.cisco.com/warp/public/614/19.html, 2005-12-12

[46] http://www.expansys.se/product.asp?code=119834, 2005-12-17.

[47] http://www.vodafonebusiness.co.uk/index.php?page=124, 2005-12-17.

[48] J. Lorchat, T. Noel, Power Performance Comparison of Heterogeneous Wireless Network Interfaces, IEEE Vehicular Technology Conference (VTC'03), Orlando, Florida USA, October 2003.

[49] M. Kazantzidis, D. Maggiorini, M. Gerla, Network independent available bandwidth sampling and measurement. *Proc. 2nd International Workshop on QoS in multiservice IP Networks* (QOS-IP 2003), Milano (Italy), February 24-26 2003.

[50] Robert Brännström. *Network-layer mobility in wireless ad hoc access networks***,** Licentiate in engineering thesis, Luleå University of Technology, December 2005. ISSN 1402-1757 / ISRN LTU-LIC-05/68-SE / NR 2005:68.

[51] http://openvpn.net, 2005-12-12

[52] http://fedora.redhat.com, 2005-12-12.

[53] http://www.ethereal.com, 2005-12-12.

[54] http://www.grid.unina.it/software/ITG, 2005-12-12.

# Appendix A

# User's Guide

## *A.1 Mobile Node*

### A.1.1 System requirements

The system requirements for the MN application are:

- Linux Platform. The MN application has mainly been tested on a laptop running the Linux distribution *Fedora core 4* [51], with kernel 2.6.11.
- Java™ 2 Runtime Environment, Standard Edition v 1.5.0 (J2SE).

The Linux kernel has to be compiled to support network devices:

- Universal TUN/TAP device driver
- PPP (point-to-point protocol)
- IP: advanced router
- IP: policy routing

The open source packages *openVPN* and *iproute* must be installed. The MN must have a public home IP address at the same local network as the HA. The MN's current IP address must also be public.

### A.1.2 Configuration

DNS-queries will be sent with the home address as source address. This requires that the MN has to configure its DNS-server appropriate for its home address. To accomplice this, the DNS file *resolv.conf*, that matches a resolv.conf file at the home network, has to be copied to the same directory as where the MN application is started. In a Linux system, the *resolv.conf* is found under the *etc* directory.

### A.1.3 Start

A user must have *superuser* rights to run this application, due to system reconfiguration scripts. The command *java SEMO* starts the application, and the GUI shown in fig.31 appears.

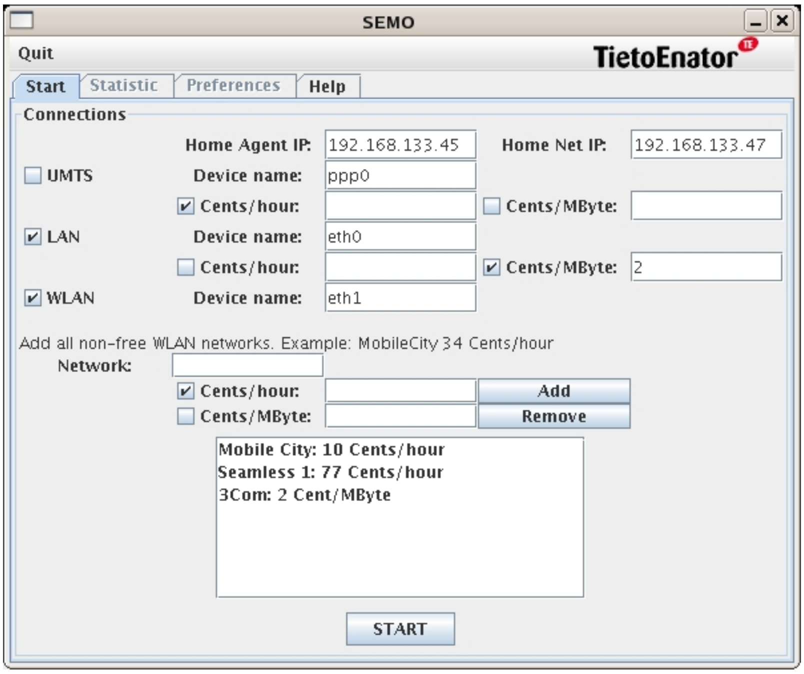


**Figure 31: Configure and start SEMO.**

As seen in the fig.31, the user can chose to utilize WLAN, LAN and UMTS access networks. The user shall indicate the following parameters:

- **Home Agent IP:** The IP address of the Home Agent.
- **Home Network IP:** The MN's IP address at the home network.
- **Device Name:** The name associated with the specific interface.
- **Cost:** The cents/hour or cents/Mbyte cost for the specific access network.

All the non-free WLAN access networks should be added to the WLAN list. Every time SEMO connects to a WLAN network, the list is traversed to see if the cost of the network known. When the *Start* button is pressed, SEMO automatically tries to establish connections to the indicated network interfaces and the *monitor* tabular, see fig.32, is displayed.

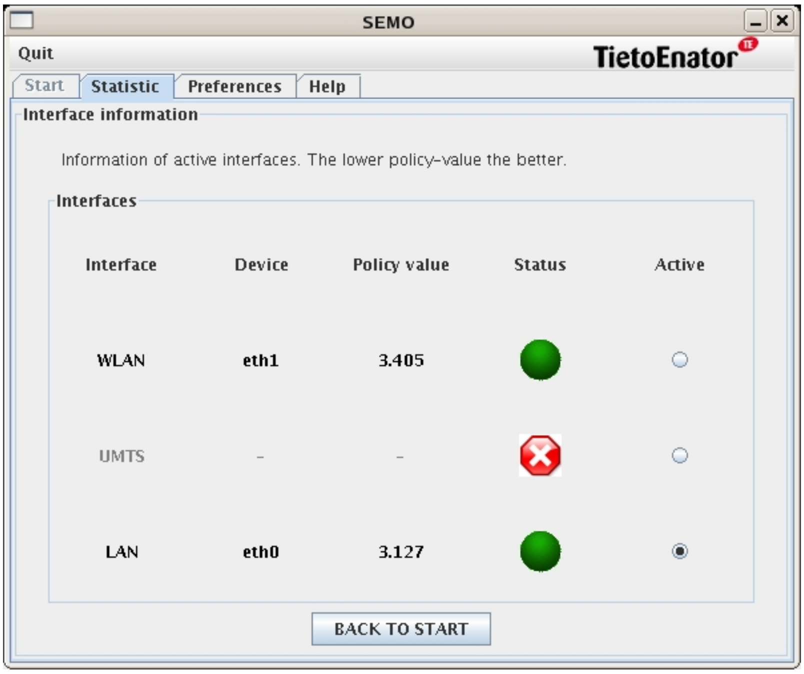


**Figure 32: Monitor network interfaces.**

This tabular present the following information for the network interfaces.

- **Interface:** Indicates the networks interface type. Available network technologies are UMTS, WLAN and LAN.
- **Device:** The device name associated with the interface.
- **Policy value:** The interface calculated policy value. The interface with the lowest policy value is considered as the most suitable interface. The value is continuously updated.
- **Status:** A green icon indicates that the connection to the specific interface is up, whereas a red icon indicates that no connection exists.
- **Active: I**ndicates which interface that is currently utilized for traffic.

To appoint the most suitable interface, SEMO's policy uses three QoS parameters: cost, battery and capacity. These parameters are initially configured to be equally prioritized. The user can reconfigure the priority of the parameters via the *Preferences* tabular, see fig.33.

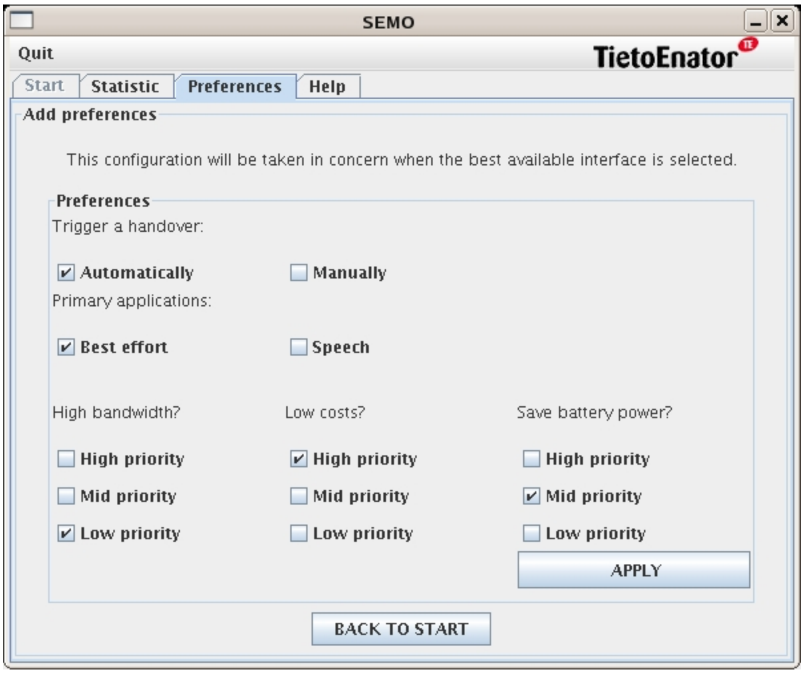


**Figure 33: Set user preferences.**

A user can indicate if a handover should be triggered automatically or manually. Furthermore, if the speech box is enabled, SEMO will avoid handovers to networks with high RTT, unless no other access network is available. When the apply button is pressed, specified preferences will in real time affect the policy decision concerning which interface that is most suitable. The application's system requirements can be found under the *Help* tabular, see fig.34.

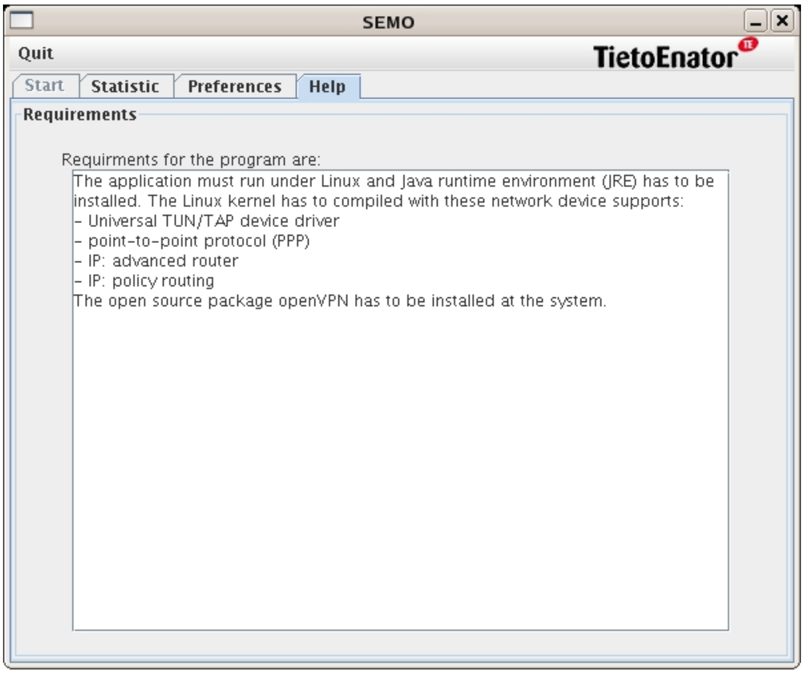


**Figure 34: Program help.**

## *A.2 Home Agent*

### A.2.1 System requirements

The system requirements for the HA application are:

- Linux Platform. The HA application has mainly been tested on a laptop running the Linux distribution *Fedora core 4*, with kernel 2.6.11.
- Java™ 2 Runtime Environment, Standard Edition v 1.4.2 (J2SE).

The Linux kernel has to be compiled to support Universal TUN/TAP device driver and *openVPN* has to be installed. The IP address that the HA uses must be public.

### A.2.2 Configuration

Before starting the HA application, a user has to specify the name of the currently utilized network device and its associated MAC address. This is done in the file *ha.config*. For example, if the computer uses a LAN networks card bound to eth0 with MAC address 00:C0:4F:A9:1E:EA, *ha.config* should contain this information:

```
Interface eth1
Mac 00:C0:4F:A9:1E:EA
```

### A.2.3 Start

A user must have *superuser* rights to run this application, due to system reconfiguration scripts. The command *java HomeAgent* starts the application.