Fast handover and QoS framework for mobility support in heterogeneous networks

Mai, Ngoc Son

2008

Mai, N. S. (2008). Fast handover and QoS framework for mobility support in heterogeneous networks. Master's thesis, Nanyang Technological University, Singapore.

<https://hdl.handle.net/10356/18858>

<https://doi.org/10.32657/10356/18858>

NANYANG
TECHNOLOGICAL
UNIVERSITY

Fast Handover and QoS Framework to support Mobility

A Thesis

Submitted to the School of Computer Engineering
of the Nanyang Technological University

by

Mai Ngoc Son

Supervisor: A/P Yeo Chai Kiat

in partial fulfillment of the requirement for the degree of
Master of Engineering

2008

FAST HANDOVER AND QoS FRAMEWORK FOR MOBILITY SUPPORT IN HETEROGENEOUS NETWORKS

MAI NGOC SON

School of Computer Engineering

A thesis submitted to the Nanyang Technological University
in partial fulfillment of the requirement for the degree of
Master of Engineering

2008

Abstract

The next generation wireless network is predicted to be a heterogeneous network of various wireless technologies such as: WiFi, WiMAX, Bluetooth, GPSR, and UMTS. With these wireless technologies, the Internet users can virtually access the Internet from anywhere and with a number of options; either at home by WiFi or outdoor by WiMAX and UMTS. However, the current network architecture does not support mobile devices to roam freely across networks or in other words, does not have mobility support. A mobile device can not keep its on-going sessions continued while crossing to another network. There have been several proposals to extend mobility support to the current IP protocol. Besides the standard Mobile IP, Terminal Mobility Support Protocol is another proposal to support application mobility using Session Initiation Protocol. Even though these proposals allow mobile nodes to roam freely across different networks, their poor handover management still poses some problems such as: long handoff delay, high packet loss, lack of QoS provisioning, no efficient support to handoffs between heterogeneous networks.

In this thesis, we propose a link-layer handover management scheme, *Virtual Handover Management* (VHM), which is an extension to the existing mobility support protocol, to accommodate fast and seamless handoffs across heterogeneous networks. The Virtual Handover Management includes the *Virtual Device Driver* (VDD) and the *Virtual Auto Handoff* (VAH). Virtual Device Driver is a module in between Link layer and Network layer. It controls the wireless access interface to use, namely WiFi, Bluetooth, or 3G. On the other hand, Virtual Auto Handoff comprises a set of daemons monitoring and making handoff decision on which wireless network. VAH communicates with VDD to execute handovers. VHM employs proactive handover method; hence the handovers are fast and uninterrupted. VHM supports most emerging wireless technologies such as:

Bluetooth, WiFi, WiMAX, GPRS and UMTS. VHM is integrated successfully to TMSP, resulting in *TMSP-VHM*, to give a complete solution to mobility across heterogeneous networks.

Various wireless technologies exhibit differences in bandwidth, delay and coverage; this raises the issue of maintaining users' perceived quality-of-service (QoS) across these technologies. Even though the availability of multiple technologies increases the coverage, the variability of bandwidth and delay introduces difficulties for applications to run while maintaining the QoS. This QoS issue is important in considering the growth of mobile users and real-time applications in this mobile environment. In this thesis, we also propose a QoS framework in a mobility scenario to cater for QoS-aware applications so that they can adapt to the variability of bandwidth and delay across different technologies, and also across different base stations. This QoS framework is built based on the TMSP-VHM. Through our framework, end-to-end QoS signaling is facilitated between the terminals; adaptive applications can thus take advantage of this signaling to maintain users' perceived QoS.

Acknowledgments

I would like to thank my supervisor Associate Professor Yeo Chai Kiat for her generous guidance, support, and encouragement during this research project.

I would like to thank my co-supervisor Associate Professor Lee Bu Sung Francis for his support and encouragement.

I would like to express my gratitude to Dr. Lim Teck Meng and Dr. Juki Wirawan Tantra, for giving me useful guidance, support and discussions. I would like to thank my colleague Mr. Dang Duc Nguyen and Mr. XiaYang for their generous help during the project and the writing of this thesis.

I would like to thank my family and my friends for their love and continuous support. And lastly, my appreciations go to the lab technicians at Centre for Multimedia and Network Technology (CeMNET) and other research officers at CeMNET.

Contents

Abstract	i
Acknowledgments	iii
List of Figures	vi
List of Tables	x
1 Introduction	1
1.1 Wireless Technologies	1
1.2 Fourth-Generation Communications System	6
1.3 Motivation and Objectives	6
1.4 Major Contributions of the Thesis	8
1.5 Organization of the Thesis	9
2 Literature Review	10
2.1 IP Mobility Support - Existing Solutions	10
2.2 Handover Management	18
2.3 Mobility Support in IPv6	22
2.4 QoS in Heterogeneous Networks	24
2.5 Summary	26
3 Virtual Handover Management in IPv4	27
3.1 Proposed Virtual Handover Management	29
3.1.1 Virtual Device Driver	30
3.1.2 Virtual Auto Handover	34
3.1.3 VHM Integration into Mobility-Support TMSP	40
3.2 Virtual Handover Management Implementation	42

3.3	Experimental Result and Discussion	46
3.3.1	Reactive versus Proactive Handover Methods	48
3.3.2	Vertical Handover across Heterogeneous Networks	53
3.4	Summary	65
4	Virtual Handover Management in IPv6	66
4.1	Virtual Handover Management Implementation	67
4.2	Experimental Result and Discussion	70
4.2.1	Reactive versus Proactive Handover Methods	70
4.2.2	Vertical Handover across Heterogeneous Networks	74
4.3	Summary	78
5	Quality of Service in Handover Management	79
5.1	Proposed QoS Framework in TMSP-VHM	79
5.1.1	End-to-end SIP Application Signaling	81
5.1.2	End-to-end Network QoS Signaling	84
5.2	QoS Framework Implementation	86
5.3	Experimental Result and Discussion	90
5.3.1	Throughput Adaptation	91
5.3.2	Jitter Adaption	95
5.4	Summary	99
6	Conclusions and Future Work	100
6.1	Conclusion	100
6.2	Future Work	102
6.3	Summary	103
	Author's Publications	104
	References	105

List of Figures

1.1	Wireless Network Technologies.	2
1.2	Illustration of a Bluetooth PAN.	5
1.3	Illustration of the 4 th Generation of Wireless Networks.	7
2.1	Illustration of MIP operation when the mobile node is on a foreign network.	12
2.2	Session handover with reactive handover.	16
2.3	Illustration of IP redirection scheme in TMSP. O-s and O-d denote the original CoAs of the source and the destination respectively. N-s and N-d denote the current CoAs of the source and the destination respectively.	17
2.4	TMSP and MSIP Mobility Management.	18
2.5	Handoffs across homogeneous and heterogeneous networks.	20
3.1	Virtual Device Driver in IP/TCPv4 protocol stack.	30
3.2	Interface-switching from WiFi interface to 3G interface.	31
3.3	Reactive and Proactive handover methods.	33
3.4	Connection State Diagram of a mobile device with 3 wireless access interfaces: Bluetooth, WiFi and 3G.	37
3.5	Flowchart of the network selection procedure when the mobile node is in 3G network.	38
3.6	Flowchart of the network selection procedure when the mobile node is in WiFi network.	39
3.7	Experiment Testbed	46
3.8	Without VHM: Throughput received at the mobile node when the correspondent node transmits at the rate of 768 Kbps.	49

3.9 With VHM: Throughput received at the mobile node when the correspond-
ent node transmits at the rate of 768 Kbps. 50

3.10 CDF of handover latencies with and without VHM integrated in TMSP . 51

3.11 3G-WiFi testbed layout 54

3.12 Bluetooth-WiFi: Throughput received at the mobile node when the cor-
respondent node transmits at the rate of 768 Kbps. 55

3.13 Bluetooth-WiFi: Throughput received at the mobile node when the cor-
respondent node transmits at a high rate of 4Mbps. 55

3.14 3G-WiFi: Throughput received at the mobile node when the correspon-
dent node transmits at half of the rate of 768 Kbps. 56

3.15 3G-WiFi: Throughput received at the mobile node when the correspon-
dent node transmits traffic at a rate of 1 Mbps. 56

3.16 Bluetooth-WiFi: CDF of handover delays of vertical handovers when the
correspondent node transmits at the rate of 8 Kbps. 60

3.17 Bluetooth-WiFi: CDF of handover delays of vertical handovers when the
correspondent node transmits at the rate of 64 Kbps. 60

3.18 Bluetooth-WiFi: CDF of handover delays of vertical handovers when the
correspondent node transmits at the rate of 400 Kbps. 61

3.19 3G-WiFi: CDF of handover delays of vertical handovers when the corre-
spondent node transmits at the rate of 8 Kbps. 61

3.20 3G-WiFi: CDF of handover delays of vertical handovers when the corre-
spondent node transmits at the rate of 64 Kbps. 62

3.21 3G-WiFi: CDF of handover delay of vertical handovers when the corre-
spondent node transmits at the rate of 400 Kbps. 62

3.22 Illustration of handoff from Bluetooth and WiFi and vice versa. 63

4.1 Without VHMv6: Throughput received at the mobile node when the cor-
respondent node transmits at the rate of 768 Kbps. 72

4.2 With VHMv6: Throughput received at the mobile node when the corre-
spondent node transmits at the rate of 768 Kbps. 72

4.3 CDF of handover delays with and without VHMv6. 73

4.4 Bluetooth-WiFi IPv6: Throughput received at the mobile node when the correspondent node transmits at the rate of 768 Kbps. 75

4.5 Bluetooth-WiFi IPv6: Throughput received at the mobile node when the correspondent node transmits at a high rate of 4 Mbps. 75

4.6 Bluetooth-WiFi IPv6: CDF of handover delay of vertical handovers when the correspondent node transmits at the rate of 8 Kbps. 77

4.7 Bluetooth-WiFi IPv6: CDF of handover delay of vertical handovers when the correspondent node transmits at the rate of 64 Kbps. 77

4.8 Bluetooth-WiFi IPv6: CDF of handover delay of vertical handovers when the correspondent node transmits at the rate of 400 Kbps. 78

5.1 Description of session handover protocol and QoS signaling in proactive handover. 82

5.2 Description of session handover protocol and QoS signaling in reactive handover. 82

5.3 Illustration of the network infrastructure considered in the mobility scenario. 86

5.4 Illustration of QoS Information exchange between TMSP agent and adaptive applications. 89

5.5 Without QoS Framework: The videoconferencing application at the correspondent node is not able to match the mobile node’s current link access capability. 92

5.6 Without QoS Framework: The throughput received at the mobile node is distorted with high percentage of lost traffic. 92

5.7 With QoS Framework: The adaptive videoconferencing application at the correspondent node dynamically changes media compression-rate to match the mobile node’s current link access capability. 93

5.8 With QoS Framework: The throughput received at the mobile node when the adaptive videoconferencing application at the correspondent node dynamically changes compressed bit rate to match the current mobile node’s link access capability. 93

5.9 With QoS Framework: The Interarrival Jitter when the correspondent node transmits at the rate of 768 Kbps when the mobile node is at the WiFi network and at the rate of 384 Kbps when the mobile node is at the 3G network. 96

5.10 Without QoS Framework: The Interarrival Jitter when the correspondent node transmits at the rate of 768 Kbps when the mobile node is at the WiFi network and the 3G network. 97

5.11 Illustration of effects of compression rate on the interarrival jitter. 97

List of Tables

1.1	Wireless Technologies' Specifications.	2
1.2	Different versions of IEEE 802.11 standard.	3
3.1	Link Event Table.	35
3.2	Command Types.	35
3.3	Types of Handovers supported by original TMSP and TMSP-VHM. . . .	41
3.4	Triggering Thresholds for Link Events.	44
3.5	Sample format of a triggering LU event from interface ath0, using NTUWL Access Point.	44
3.6	Sample format of a command from main daemon to bring down the inter- face ath0.	45
3.7	Experiment Components	47
5.1	SIP REGISTER message with QoS.	87
5.2	SetQoS message.	88
5.3	SIP INVITE message with QoS.	88

Chapter 1

Introduction

1.1 Wireless Technologies

Wireless networks are becoming more pervasive with a number of new wireless communication technologies, inexpensive wireless equipment, and broader access availability. More and more people become active users of wireless networks and these networks are transforming the way people use computers, cell phones, and other Personal Digital Assistant at work, home, and when traveling. Not satisfied with that, the wireless world continues to grow faster as more robust technologies are developed to free us from wires for greater simplicity, convenience and efficiency.

A variety of wireless access technologies are available nowadays. These include: Wireless Local Area Network (WLAN), Worldwide Interoperability for Microwave Access (WiMAX), General Packet Radio Service (GPRS), and Bluetooth Personal Area Network (PAN). Figure 1.1 and Table 1.1 summarizes these wireless technologies' specifications in terms of range and speed. In the following section, we discuss in detail the most emerging wireless technologies.

WiFi, or IEEE 802.11 standards [1], is a wireless technology for wireless local area network (WLANs). The current generation of WiFi supports data rate up to 54Mbps within 140 meters of the access point. Table 1.2 summarizes the different versions of IEEE 802.11 standard [2]. WiFi is usually deployed in a distributed way to offer last

Figure 1.1: Wireless Network Technologies.

Table 1.1: Wireless Technologies’ Specifications.

Technology	PAN	LAN	MAN	WAN
Standards	IEEE 802.15 Bluetooth	IEEE 802.11 WiFi	IEEE 802.16 WiMAX	3G UMTS
Speed	max 3Mbps	11-54Mbps	11-100Mbps	max 14.4Mbps
Range	Short 10 meters	Medium 100 meters	Medium-long 50 kilometers	Long city-range

few-hundred-meter connectivity to campus, organization networks or to the Internet. For example, NTU has deployed its *NTUWL* wireless network that allows students to access the Internet wirelessly within the campus. In Singapore, residents can soon enjoy free unlimited WiFi services in many public areas within the island with *Wireless@SG* program¹.

Although each WiFi access point can support connections only over a range of a hundred meters, it is possible to provide contiguous coverage over a wider area by using multiple access points. A number of organizations and university campuses have deployed

¹Wireless@SG is a wireless broadband programme developed by IDA as part of its Next Generation National Infocomm Infrastructure initiative. It will be run and developed in the next two years by three local wireless operators who will deploy a wireless broadband network in Singapore. Users can enjoy free, both in-door and outdoor seamless wireless broadband access with speeds of up to 512 Kbps at most public areas [3].

Table 1.2: Different versions of IEEE 802.11 standard.

Version	Release Date	Data Rate		Range
		Max	Typical	
802.11a	October 1999	54 Mbps	23 Mbps	approx. 120 m
802.11b	October 1999	11 Mbps	4.5 Mbps	approx. 140 m
802.11g	June 2003	54 Mbps	19 Mbps	approx. 140 m
802.11n	est. June 2009	248 Mbps	74 Mbps	approx. 255 m
802.11y	est. June 2008	54 Mbps	23 Mbps	approx. 5000 m

such contiguous WiFis (e.g. *NTUWL*). However, the WiFi technology is still not designed to support efficient handoffs when users move between access point coverage areas (this mobility problem is addressed by the cellular technology). Even though WiFi is generally supporting data communications (in contrast to cellular networks which are mainly for voice traffic), it is possible to support multimedia services over WiFi with the growing interest in supporting real-time services such as voice and video over Internet Protocol (IP) networks.

WiMAX [4], Worldwide Interoperability for Microwave Access, is a wireless metropolitan area network technology (WMAN). WiMAX is specified in IEEE 802.16 standard. It enables the delivery of last mile wireless broadband access as an alternative to wired broadband like cable and DSL. WiMAX provides fixed, nomadic, and portable and, soon, mobile wireless broadband connectivity without the need for direct line-of-sight with a base station. In a typical cell radius deployment of three to ten kilometers, WiMAX Forum Certified systems can be expected to deliver capacity of up to 40 Mbps per channel, for fixed and portable access applications. The bandwidth is sufficient to simultaneously support hundreds of mobile devices with T-1 speed connectivity and thousands of residences with DSL speed connectivity. Mobile network deployments are expected to provide up to 15 Mbps of capacity within a typical cell radius deployment of up to three kilometers. WiMAX technology will soon be incorporated in notebook computers and PDAs by 2008 [4].

CHAPTER 1. INTRODUCTION

The main differences between WiFi and WiMAX are speed and coverage. WiFi has a typical bandwidth of 54 Mbps whereas WiMAX can have a bandwidth of up to 75 Mbps. However, the real advantage of WiMAX over WiFi is not bandwidth; it is the coverage range. Whereas WiFi operates at service radii of about 100 meters, WiMAX can deliver high-bandwidth connectivity to devices as far as 10 kilometers away.

3G is the third generation of mobile phone standards and technology, after 2G. 3G networks are wide area cellular telephone networks which evolved to incorporate high-speed internet access and video telephony. 3G technologies enable network operators to offer users a wider range of more advanced services while achieving a greater network capacity through improved spectral efficiency. Services include wide-area wireless voice telephony and broadband wireless data, all in a mobile environment. The most significant feature of 3G mobile technology is that it supports greater numbers of voice and data customers, especially in urban areas, and higher data rates at a lower incremental cost than 2G.

In early 1999, Singtel started a 3G service using Wideband-CDMA (W-CDMA) technology with leading Japanese operator, NTT DoCoMo, and the Centre for Wireless Communications, National University Singapore. Mobile operator M1 of Singapore has also launched its 3G/W-CDMA networks in 2005.

Unlike WiFi networks, which are short range, high-bandwidth, and primarily developed for data, 3G networks are long range, lower bandwidth and preferably for voice data.

Bluetooth [5] is a common standard for wireless technology developed for the initial purpose of replacing cables between electronic devices located in the same area. Bluetooth is designed to be short range connectivity (approximately 10 meters for class 2, and 100 meters for class 1), and low bandwidth (1 Mbps for version 1.2, and 3 Mbps for version 2.0). Owing to its low manufacturing cost and low power consumption, Bluetooth is

CHAPTER 1. INTRODUCTION

readily found in all types of mobile devices such as PDAs, laptops, mobile phones, to provide wireless connection capability. It is, however, rarely used for general-purpose WPAN networking due to its short range and low bandwidth. Bluetooth wireless technology is geared towards voice and data applications

Figure 1.2: Illustration of a Bluetooth PAN.

Even though Bluetooth is preferably used to enable cable-free connections between electronics devices, it can be used for wireless network access. The IEEE 802.15 project [5] has derived a Bluetooth Wireless Personal Area Network (WPAN) standard. IEEE 802.15 WPAN is a high-data-rate standard and is designed to provide sufficient quality of service for real-time applications such as VoIP, home multimedia video and music.

Figure 1.2 illustrates a Bluetooth piconet, a Personal Area Network. A Bluetooth device acts as a network base station. It then allows other Bluetooth-enabled devices to join a piconet and access LAN resources. The Bluetooth base station functions as a router that enables connection between Bluetooth-enabled devices and a LAN or the Internet. Typically, a Bluetooth base station is designed to serve small computing devices that require network access with low power consumption such as PDAs and mobile phones. But, it may also support laptop or desktop computers and other Bluetooth-enabled devices.

1.2 Fourth-Generation Communications System

Every wireless technology has its own pros and cons. Bluetooth technology is ideal for small personal home networks where coverage is only a few meters. The Bluetooth networks are low-cost, less power-hungry and very convenient for home multimedia entertainment or simple file sharing for small handheld devices. On the other hand, WiMAX is mostly used to provide last-mile Internet access to rural and metropolitan area. WiFi technology can cover areas of only a few thousand square meters, making them suitable for organization networks and public hot-spots such as hotels and airports. On the contrary, 3G mobile networks require significant capital investments, support limited peak rates, but offer a much wider area of coverage. In comparison, 3G networks' coverage is more extensive; and 3G networks enable ubiquitous connectivity with greater mobility to users than WiMAX and WiFi. In general, notebook users prefer WiFi/WiMax network access; while mobile and PDA users would still opt for 3G or HSDPA.

The wireless access technologies offer complementary solutions and characteristics. This would lead to the current situation of multiple technologies coexisting; and future mobile communication systems will be built upon wireless overlay networks of heterogeneous technologies. This trend is going to form the 4th generation wireless networks. This promising trend is currently recognized and favored by the wireless technology development community. Figure 1.3 illustrates the scenario of 4th generation of wireless networks where Bluetooth, WiFi, WiMAX and 3G all exist.

1.3 Motivation and Objectives

4G, an acronym for *Fourth-Generation Communications System*, is a term used to describe the next generation of wireless communications. 4G would be IP-based, packet switched networks of heterogeneous technologies. 4G is expected to provide pervasive

Figure 1.3: Illustration of the 4th Generation of Wireless Networks.

coverage, ubiquitous wireless access and high-rate data services with QoS provisioning for multimedia applications. Although there is no formal definition of 4G, there are certain issues which need to be addressed for 4G to achieve these objectives.

The first problem is how to accommodate fast and uninterrupted handoffs across heterogeneous networks. With the coexistence of various access technologies, there are two types of network handovers in 4G: horizontal and vertical handovers. Horizontal handover occurs when mobile devices move across networks of the same technologies (e.g. from one WiFi network to another); whereas vertical handovers refers to those movements across heterogeneous-technology networks (e.g. from a WiFi network to 3G network). A number of studies and schemes have been proposed in the literature for horizontal handovers. However, there are a limited number of studies on vertical handoffs, leaving many issues to be addressed in order to provide seamless roaming experience for mobile users.

The second problem is how to support Quality of Service for next generation multimedia applications, such as real time, video conferencing, video phone, movie streaming,

CHAPTER 1. INTRODUCTION

---

online gaming, and mobile TV. Various wireless access technologies exhibit different characteristics in bandwidth, delay and coverage. This raises the issue of maintaining users' perceived quality-of-service (QoS) when moving across these technologies. This QoS issue is important considering the growth of mobile users and real-time, multimedia applications in the current mobile environment.

This master project aims to tackle the vertical handoff and QoS provisioning issues in 4G networks. We propose a handoff scheme with integrated QoS framework to accommodate fast, seamless and smooth roaming across heterogeneous networks. We also address the QoS provisioning problem with our proposed QoS framework.

1.4 Major Contributions of the Thesis

The main contributions of this thesis are as follows:

- (i) The thesis introduces a link-layer handover management scheme for mobile devices to roam freely across networks of *heterogeneous technologies* by supporting multiple access interfaces. The handover management scheme supports an *uninterrupted* and *fast* network-handover experience when mobile devices handoff between networks. The handover management supports both current IPv4 and future IPv6 network.
- (ii) The thesis introduces a solution to QoS provisioning in heterogeneous networks with a QoS framework. The QoS framework provides end-to-end QoS signaling for adaptive applications to dynamically change their attributes according to the capability of the current access connection. The QoS framework is also integrated with another QoS reservation scheme to support advanced network reservation. With our QoS framework, mobile devices are able to experience **smooth** transition across networks of heterogeneous types.

1.5 Organization of the Thesis

The rest of the thesis is organized as follows: Chapter 2 presents some of the previous works on IP mobility. Chapter 3 presents the Virtual Handover Management (VHM). VHM is an extension to the mobility support protocols to enable fast and smooth handovers across both homogeneous and heterogeneous wireless networks. Chapter 4 presents the Virtual Handover Management in IPv6 (VHMv6). VHMv6 employs the same idea as VHMv4 and successfully extends the solution to IPv6. Chapter 5 discusses the proposed QoS framework to address the QoS provision problem. Chapter 6 concludes the thesis and discusses the future works.

Chapter 2

Literature Review

2.1 IP Mobility Support - Existing Solutions

Cellular network provides voice and data services to users on the move. To deliver services to the mobile users, the cellular network is capable of tracking the locations of the users, and allowing user movement during the conversations. Original design of IP-based networks, however, does not have mobility support. In IP-based networks, each node is given a unique IP address. The IP address uniquely identifies the node's point of attachment to the Internet; hence defining how the node is reachable. When the node changes its access networks, which usually results in a change of the node's IP address, it is impossible for the node to maintain transport and higher-layer sessions. Therefore, a new, scalable mechanism is required for supporting node mobility within the Internet. There is a number of mobility management protocols proposed to add mobility support to IP-based networks, such as: Mobile IP [6] and TMSP [7]. Mobile IP provides an efficient, scalable mechanism for roaming within the Internet. Mobile IP requires each mobile node to have a unique, permanent IP address. Hence, nodes may change their point-of-attachment to the Internet without changing their IP addresses. This allows them to maintain transport and higher-layer connections while on the move. On the other hand, TMSP enables node mobility through the use of an IP-to-IP address mapping at the network layer, and a SIP signaling services to manage changes in IP address.

CHAPTER 2. LITERATURE REVIEW

Different protocols employ different methods to support mobility. At the core, however, these protocols achieve mobility by two basic functions: (1) Handoff Management and (2) Location Management. Location management is a process by which the network updates the location database and supports location/redirect services to authorized users and authorities. A Location Management scheme provides a means for networks to discover the whereabouts of a mobile host and to deliver a call/session to its correspondent host. The basic mechanism required for a Location Management scheme is location registration/update and session delivery. Handoff Management is a process that allows an established call/session to continue when a mobile node moves from one network to another network without interruptions in the call/session. In more detail, Handoff management comprises several operations, namely: Network Discovery, Handover Decision and Handoff Execution. The following sections describe Mobile IP and TMSP in more details.

Mobile IP (MIP) [6] is an Internet Engineering Task Force (IETF) standard communications protocol that is designed to allow mobile device users to move from one network to another while maintaining a permanent IP address and ongoing connection. This is how Mobile IP works. A mobile node is given a long-term IP address on a home network. When the mobile node is on its home network, it operates without mobility service. When the mobile node moves to a foreign network, it obtains a care-of address (CoA) from the *foreign agent*. The mobile node then registers its new Care-of-Address (CoA) with its *home agent*. Here, MIP provides the Location Management service through the home agent and foreign agent. Packets sent to the mobile node's home address are intercepted by its home agent, and then tunneled by the home agent to the mobile node's CoA. In the reverse direction, packets sent by the mobile node are directly delivered to their destination using standard IP routing mechanism, bypassing the home agent. In this way, the ongoing communication between the mobile node and its correspondent nodes

CHAPTER 2. LITERATURE REVIEW

does not break when the mobile node moves. Here, MIP provides Handoff Management services via tunneling technique and the mobile node's permanent IP address. Figure 2.1 describes how the traffic flows when the mobile node is away from the home network:

- (1) Traffic from the correspondent (CN) node to the mobile node (MN) is routed to the mobile node's Home Agent.
- (2) Packets are intercepted by the Home Agent and tunneled to the CoA of the mobile node.
- (3) The Foreign Agent de-tunnels packets and forwards to the mobile node.
- (4) Traffic from the mobile node to the correspondent node is routed directly without tunneling, by standard IP routing.

Figure 2.1: Illustration of MIP operation when the mobile node is on a foreign network.

Mobile IP has several drawbacks, which are: added communication latency, single point of failure, scalability and handoff latency. Firstly, Mobile IPv4 does not optimize the traffic routes; it adds inefficient latency to the communication between the mobile node and the correspondent node. When the mobile node is on a foreign network, almost half of the traffic between the mobile node and the correspondent node are not direct and has to go through the home agent. This means that even if both the mobile node and the correspondent node are in Changi Airport, but the home agent is in Nanyang

CHAPTER 2. LITERATURE REVIEW

Technological University (NTU), the traffic from the correspondent node must go all the way from Changi Airport to NTU and back to Changi Airport, which is the current location of the mobile node. Secondly, Mobile IP has Home Agent as single point of failure. Its functionality depends completely on connectivity of the home agent. If a mobile node loses contact with its home agent, say due to the home agent's network failure, it will also lose current connection with the correspondent node even if direct communication with the correspondent node would still be possible. Thirdly, scalability is an issue. Since traffic from the correspondent node to the mobile node must go through the home agent, the home agent must support every mobile-node-correspondent-node pair in terms of bandwidth, processing capacity and memory for redirection lookup tables. This limits the number of mobile nodes a home agent can support. The last drawback is the handover latency. MIP handover time is the sum of new link association time (Layer-2 association time + Layer-3 IP address assignment time) plus the round-trip time to the home agent. L2 association should be in milliseconds. L3 association, which acquires new IP address through, say DHCP server, might take longer, in approximately 1 second. This latency is definitely not acceptable for real-time applications such as video conference, VoIP.

Terminal Mobility Support Protocol (TMSP) is proposed in [7]. TMSP supports IP mobility using Session Initiation Protocol (SIP) [8]. TMSP is a terminal-based IP mobility support protocol. It does not require mobile nodes to have permanent IP addresses. It provides IP mobility for all data connection in a mobile node by using a Layer-3 IP-to-IP address mapping method. This method ensures the IP address transparency for the protocol layers above the network layer. It uses a user agent (UA) to manage the change in IP address. It relies on a dynamic directory service, SIP Redirect Server (SIP-RS), to provide an URI-to-IP¹ address resolution. Here, TMSP provides

---

¹URI stands for SIP Uniform Resource Identifier.

CHAPTER 2. LITERATURE REVIEW

---

location management service through SIP signaling service, and handoff management service through IP-to-IP address mapping technique.

There are 3 basic operations to manage IP mobility: (1) session initiation, (2) session handover, and (3) IP redirection. Session initiation operation is started when a node, the mobile node or the correspondent node, tries to communicate to another one, either mobile or stationary. Session handover is executed when a mobile node moves to another network, either because it finds a better connection to use or its current connection is disrupted. IP redirection involves all the packets sent by both the mobile node and the correspondent node; basically, TMSP performs IP redirection so that any packet is routed properly to the current CoA of the mobile node.

(1) Session initiation. Most of the IP mobility scheme (e.g. Mobile IP) relies on a directory service for the mobile nodes to be contactable; TMPS is not an exception. In TMSP, mobile nodes that require mobility support, have to register with a SIP server and are assigned unique SIP Uniform Resource Identifiers (URIs). Here the SIP server acts as a directory service for the mobile nodes. This directory service is a necessity for the mobile nodes to be contactable by other nodes, as IP address is location and provider dependent. Compared to Mobile IP, which requires home and foreign agents, TMSP only requires a SIP server, which is widely available as part of the current Internet infrastructure. When a mobile node wants to connect to a correspondent node, the TMSP module in the mobile node first queries the SIP server for the correspondent node's current CoA using the correspondent node's URI. This is similar to how DNS currently works, which is transparent to the application that initiates the connection. After this, the mobile node informs the correspondent node of the TMSP connection and passes the mobile node's URI to the correspondent node for session management use. The correspondent node will thus reply to the mobile node acknowledging the connection. This two-way handshake is also useful for information exchanges between the mobile node and the correspondent

CHAPTER 2. LITERATURE REVIEW

---

node. Both the mobile node and the correspondent node keep the connection data in their respective tables, which consist of the correspondent node's URI and CoA, and the connection port number. In TMSP design, both the CoA and the port number are used to identify a TMSP connection between a mobile node to a correspondent node. TMSP uses the *SIP REGISTER* and *SIP INVITE* messages for establishing connection to the SIP server and to the correspondent node, respectively.

(2) Session handover. There are two distinct handover scenarios: (1) proactive handover, and (2) reactive handover. In the first scenario, the active connection is not broken when a mobile node decides to switch its connection interface. A mobile node might decide to switch its connection when it detects quality degradation on its active connection and there is a better connection available on another interface. Thus, the mobile node must have multiple interfaces to implement proactive handover. In reactive handover, the mobile node detects a lost connection before it finds another connection and executes the handover operation. TMSP specifies only reactive handover. Let us consider the steps through reactive handover scenario. When the mobile node detects a broken connection, it first finds another access point to connect to. After the new connection is ready, the mobile node informs the SIP server of its new CoA through *SIP REGISTER* message. Then, the mobile node establishes a new TMSP connection to the correspondent node and informs the correspondent node to update the old connection to the new connection. On receiving this update message, the correspondent node first verifies the originator of the new connection by querying the SIP server of the mobile node's new CoA using the stored mobile node's URI. After the verification is completed, the correspondent node updates its tables reflecting the changes in the mobile node's CoA. Figure 2.2 illustrates the reactive handover procedure. The reactive handover procedure also needs to handle the case where both the mobile node and the correspondent node change their CoAs concurrently. When both the mobile node and the correspondent node

CHAPTER 2. LITERATURE REVIEW

change their CoAs, it is not possible for either the mobile node or the correspondent node to inform the other about the change. Here, we implement a timeout mechanism; when the mobile node that is trying to update the correspondent node with its new CoA does not receive a reply, the mobile node will enquire the SIP server for the correspondent node's new CoA. Having this new CoA, the mobile node can continue the normal reactive handover procedure described above.

Figure 2.2: Session handover with reactive handover.

(3) **IP redirection.** While MIP requires additional infrastructure support for the home and foreign agents, TMSP supports IP mobility through an IP-swapping technique on the mobile nodes themselves. The IP-swapping module is installed in the mobile nodes, thus it does not require any modification to the network operation. Assuming that the connection is already established, the operations of the IP-swapping module are as follows. TMSP intercepts all the packets sent or received by the mobile nodes. The TMSP IP redirection module first checks whether a packet needs to be processed by matching the packet's IP addresses and port number to the known TMSP connections in its table. Matching connections will thus be processed by the TMSP IP redirection module. For incoming packets, the IP addresses in the packets are modified to the original CoAs kept in the TMSP reference table. The table entry is matched for both

CHAPTER 2. LITERATURE REVIEW

the IP addresses and the port number in the packet. After the packet is processed, it is passed up to the upper layer. For outgoing packets, the IP addresses in the packet are modified to the current CoAs kept in the table. This modification thus enables the packets to be routed directly to the correspondent node's current CoA without any additional network support. We can see that using this method, the applications require no modifications; hence, TMSP provides mobility support to all existing applications. Figure 2.3 illustrates the modification of the IP addresses to the current CoAs in the sender, and the modification of the IP addresses to the original CoAs in the recipient.

Figure 2.3: Illustration of IP redirection scheme in TMSP. O-s and O-d denote the original CoAs of the source and the destination respectively. N-s and N-d denote the current CoAs of the source and the destination respectively.

Compared to MIP, TMSP has more advantages. Firstly, TMSP only requires a SIP server, which is widely and readily available. MIP, on the other hand, requires permanent IP address for each mobile node, extra Home Agents and Foreign Agents. Secondly, MIP adds overhead to all packets because it uses IP-in-IP encapsulation for packet tunneling. TMSP uses IP swapping technique, which does not increase the packet size. Thirdly, MIP does not scale very well in terms of the number of mobile nodes a home agent can support. And finally, in MIP, when the mobile node is away from its home network, its

communication with correspondent nodes is not direct, but through IP tunnel from the home agent to the foreign agent. This causes unnecessary delay in the traffic, and even worse, causes triangular routing.

2.2 Handover Management

MIP and TMSP provide mobility support for mobile devices when roaming across IP-based networks. They achieve mobility by two basic functions: 1) Handoff Management and 2) Location Management. MIP and TMSP implement Location Management through their respective directory services: TMSP uses SIP service while MIP uses the *Home Agent* and *Foreign Agent*. Both protocols, however, provide limited handoff management services. The *horizontal* handovers done by TMSP and MIP introduce long handoff latency. Moreover, their handoff managements concentrate on upper network layers, such as IP, UDP/TCP and application layers as seen in Figure 2.4, without attempting to address the Link layer mobility. Hence *vertical* handoffs across heterogeneous networks are left unfulfilled. For example, a handover from WiFi network to 3G network requires a link-layer mobility mechanism to re-establish link-layer connectivity to the node in each new location.

Figure 2.4: TMSP and MSIP Mobility Management.

CHAPTER 2. LITERATURE REVIEW

Fourth-Generation Communications System would integrate heterogeneous wireless technologies, optimizing the best of each in terms of speed and coverage where it is technically and commercially viable, to provide ubiquitous communications to the mobile devices. It is important to support uninterrupted and seamless handoff management in this integrated architecture. The existing mobility support protocols, TMSP and MIP, focus on how to allow upper-layer session continuity during the handoffs; they are not sufficient to provide vertical handoff support that is transparent to the applications. The vertical handoff between different networks poses implementation issues at the network protocol level, especially for 3G networks and for WiFi, Bluetooth, and WiMAX. It would be beneficial to have a link-layer-and-network-layer handoff management scheme to support a seamless homogeneous and heterogeneous handoff management. Figure 2.4 shows the position of a handoff management scheme in the network protocol stack. Figure 2.5 illustrates a typical handoff scenario in 4G. In Figure 2.5, there are two popular networks: WiFi network and 3G network, even though in the real scenario, there may be more technologies involved such as WiMAX and Bluetooth. A mobile device is located in one of the WiFi network. Movement (1) brings the mobile device to another WiFi network, which is of the same technology; this type of handoff is horizontal; whereas, movement (2) involves two different technologies, WiFi and 3G and the handover is characterized as vertical handover.

Current IEEE 802 standards, such as WiFi, WiMAX, and Bluetooth, do not support vertical handovers. They also do not provide triggers or other services to accelerate network handovers. Moreover, existing IEEE 802 standards provide mechanisms for detecting and selecting network base stations, but do not allow for detection and selection of network base stations in a way that is independent of the network technologies.

In the network protocol stack, the network interfaces are in the Link layer. UDP/TCP sessions make use of one interface for its traffic reception and transmission. If the network

Figure 2.5: Handoffs across homogeneous and heterogeneous networks.

interface at Link layer loses the connection with its current base station, the UDP/TCP session will terminate as well. UDP/TCP sessions cannot switch from one network interface to another during the course of the session. Since horizontal handoffs involve only one wireless interface, changing the networks can be done with the aid of mobility support protocol. However, vertical handoff involves two network interfaces of different technologies and switching the interfaces always results in the UDP/TCP session termination. To provide the vertical handoffs, a virtual network interface is to be built in the IP layer. The virtual interface takes care of switching between different network access interfaces. The virtual interface in the IP layer will then choose which network access interface to use on the basis of network speed, quality of service, cost of usage and other similar criteria. The selection policies are configurable and network handoffs can be either automated or manual.

Media Independent Handover (MIH) [9] 802.21 is an IEEE emerging standard. The standard supports algorithms enabling seamless handovers between networks of the same technologies as well as handovers between different network technologies or vertical

CHAPTER 2. LITERATURE REVIEW

---

handover. The standard provides information to allow handing over to and from 3G, WiFi, Bluetooth and WiMAX networks through different handover mechanisms. The key functionality provided by MIH is communication among the various wireless link layers and between them and the IP layer. The messages are relayed by the Media Independent Handover Function (MIHF) that is located in the network protocol stack between the Link layer of wireless technologies and IP layer. MIH may communicate with various protocols including TMSP and Mobile IP for mobility management, or *DiffServ* and *IntServ* for QoS. When a session is handed off from one wireless technology to another, MIH can support handovers by passing messages among the wireless technologies. Messages are of three types:

- *Event notifications* are passed from Link layer in the protocol stack to higher layers. For example, wireless link quality degradation is an event notification that is passed from the wireless link layer to the MIHF layer.
- *Commands* are passed down the protocol stack. For instance "Initiate Handover" is a command in which the base station MIHF provides the mobile device MIHF with a list of alternative base stations that it could use.
- *Information Service* is of three types. A higher layer may request information from a lower layer, e.g. the MIHF may request performance information, such as delay from the wireless layer. A lower layer may request information from a higher layer, e.g. the MIHF may request to know the Base Station ESSID² from the IP layer. Or one MIHF may request information from another MIHF, e.g. the availability of location-based services.

MIH receives the Event notification to know the status of its current wireless connection. It might then make a network handoff decision. The MIHFs communicate to

---

²ESSID stands for Extended Service Set ID. An ESSID identifies name of a wireless network

identify which base stations using which wireless technologies are within range and what QoS is available from them. MIH can also be used to pre-authenticate the mobile device with alternative potential base stations and to reserve capacity prior to handover. When a handover becomes necessary, much of the ground-work is therefore already in place and the session can be handed over with minimal delay and packet loss. Incoming packets to the mobile device that are delivered to the old base station after the handover can be forwarded via the new base station, thus further reducing packet loss.

2.3 Mobility Support in IPv6

Mobile IPv6 (MIPv6) [10] is designed to provide mobility support at the IPv6 layer in the network stack, thus providing IP mobility support to the transport and application layers. MIPv6 allows nodes to remain reachable while moving around in the IPv6 Internet. Each mobile node is always identified by its permanent home address, regardless of its current point of attachment to the Internet. While situated away from its home network, a mobile node is also associated with a Care-of-Address (CoA), which provides information about the mobile node's current location. IPv6 packets addressed to the mobile node's home address are transparently routed to its care-of address. Mobile IPv6 was published in 2004, in RFC 3775 [10].

The design of Mobile IP support in IPv6 (Mobile IPv6) benefits both from the experiences gained from the development of Mobile IP support in IPv4 (Mobile IPv4) [6], and from the efficient design of IPv6. Mobile IPv6 thus shares many features with Mobile IPv4, but is integrated into IPv6 and offers many other improvements. Here, we summarize some of the differences between Mobile IPv4 and Mobile IPv6:

- There is no need to deploy special routers as *foreign agents*, as in Mobile IPv4. Mobile IPv6 operates in any location without any special support required from the local router.

CHAPTER 2. LITERATURE REVIEW

- MIPv6 supports route optimization as a fundamental part of the protocol, rather than a nonstandard set of extensions. The route optimization feature removes the added latency problem for regular MIPv4 traffic. However, the cost of the route optimization is the additional handoff latency. Route optimization requires the mobile node to register its current binding at the correspondent node. Packets from the correspondent node can be routed directly to the care-of address of the mobile node. Typically, the handoff delay in base Mobile IPv6 route optimization is one round-trip time between the mobile node and the home agent for the home registration, one round-trip time between the mobile node and the home agent plus one round-trip time between the home agent and the correspondent node for the return routability procedure³, and one one-way time from the mobile node to the correspondent node for the propagation of the Binding Update message.
- Most packets sent to mobile nodes while away from home in MIPv6 are sent using an IPv6 routing header rather than IP encapsulation, reducing the amount of resultant overhead compared to MIPv4.
- The use of IPv6 encapsulation (and the routing header) removes the need in Mobile IPv6 to manage tunnel soft state in routers.

Terminal Mobility Support Protocol in IPv6 (TMSPv6) [12] is another mobility support protocol. TMSPv6 shares most of its ideas with its counterpart TMSP in IPv4 with further extension to take full advantage of the newly-standardized IPv6. TMSPv6 relies on a dynamic directory service mechanism, namely SIP Redirect Server (SIP-RS), to provide an URI-for-IP-address resolution. When a mobile node changes its Care-of-Address, it informs its recipients, correspondent nodes, and updates the directory

---

³The return routability procedure authorizes registrations by the use of a cryptographic token exchange [11].

CHAPTER 2. LITERATURE REVIEW

---

service. The mobile node uses an IP-to-IP address mapping scheme to provide IP address transparency and a new IPv6 header in each IP address to enable the packet to avoid ingress filtering and direct routing to its recipient. The detailed operation of TMSPv6 is described in Section 2.1.

TMSPv6 and Mobile IPv6 do not attempt to solve all general problems related to the heterogeneous wireless networks. In particular, these protocols do not solve the Link layer mobility management problem, the network service discovery and the QoS signaling for adaptive applications. Like TMSPv4 and MIPv4 cases, TMSPv6 and MIPv6 concentrate on mobility in upper network layers, such as IP, UDP/TCP and application layers as seen in Figure 2.4. They do not attempt to address the Link layer mobility. Hence detailed and efficient *vertical* handoffs across heterogeneous IPv6 networks are also left unfulfilled. In this thesis, TMSPv4 is abbreviated as TMSP; whereas TMSPv6 is always denoted TMSPv6. Similarly for MIP, MIPv4 and MIPv6 are used.

2.4 QoS in Heterogeneous Networks

The Request For Comment (RFC) 2205 [13] proposes Resource Reservation Protocol (RSVP), that prepares the resource reservation for real-time and multimedia traffic in the Internet. RSVP is however not a flexible protocol, and more suitable for fixed resource reservation. Once the network resource is reserved, it cannot be altered. In the current network scenario, RSVP is not adequate for accommodating the mobile nodes which frequently change their point of attachments. Thus, a new QoS mechanism is needed to handle the mobility of the nodes. Mobile RSVP is proposed in [14] as an extension to the RSVP. The main feature of this protocol is the ability to make advance reservations for a mobile node at locations where it may visit. MRSVP supports *active* and *passive* reservations. The mobile node makes an active reservation from its current location, and

CHAPTER 2. LITERATURE REVIEW

---

a passive reservation from the other locations that it might visit during the lifetime of the ongoing sessions.

Bongkyo Moon *et al.* propose a reliable RSVP path reservation architecture for multimedia applications in [15]. During the network handoffs, the architecture *reroutes* the current RSVP branch path at a *crossover router* in order to minimize resource reservation delay. Bongkyo Moon also showed that the scheme could provide QoS guarantee across networks through simulation analysis and examples.

Joachim Hillebrand *et al.* discuss another approach to the QoS provision issue in mobility environment in [16]. The architecture integrates the resource management with mobility management. It is based on a *Domain Resource Manager* concept and nicely supports different handover types in an integrated approach. The scheme supports anticipated handovers with pre-reservation of resources over the old network before the mobile node is attached to the new access point.

In this thesis, we propose a QoS framework to support QoS provisioning in the mobility scenario. Our framework is integrated with the mobility management protocol TMSP. Our QoS framework supports *end-to-end application QoS signaling* and *end-to-end network QoS reservation*. The end-to-end Application QoS signaling assists adaptive applications to adapt to the variability of bandwidths and delays of different networks. Whereas, the end-to-end network reservation makes sure the network resource is pre-reserved in the new network that mobile nodes will handoff to. Our scheme is unique from the above approaches because it is designed with mobility management TMSP in an integrated way. Thus, the handoff signaling and QoS signaling are combined and the total cost is reduced to a minimum. In the thesis, we also propose link-layer handover management to help make anticipated handover decision. Hence, the network handover is performed fast and smoothly with QoS maintained not only before and after the handover, but also during the handover.

2.5 Summary

The future mobile communication systems will be built upon heterogeneous wireless overlay networks. This raises the issues of how to support fast and smooth handovers across networks. In this chapter, we introduced two existing solutions to the IP mobility problem: TMSP and Mobile IP. Mobile IP is the Internet Engineering Task Force (IETF) standard protocol, which is described in IETF RFC 3344 [6]. TMSP is proposed in [7]. In comparison, TMSP enjoys advantages of simpler deployment and fewer overheads compared to Mobile IP.

TMSP and Mobile IP provide mobility support in Network and UDP/TCP layers. They do not attempt to solve the mobility support problem in the Link layer, which implies that vertical handoffs across heterogeneous networks are left unfulfilled. Furthermore, TMSP and Mobile IP provide very simple handover managements that do not allow fast and efficient network handoffs. In addition, various wireless technologies exhibit different characteristics in terms of speed and coverage range. In the mobility scenario, there should be a signaling channel for adaptive applications to inform each other of their QoS access capability changes so that the users' perceived QoS is maintained. Moreover, another signaling channel is required for applications to prepare the network resource in advance to maintain the network QoS during the duration of the sessions. The lack of an efficient handoff management and a QoS framework in the mobility support protocols are the main motivations for the research work in this thesis.

Chapter 3

Virtual Handover Management in IPv4

TMSP is designed for mobility support and it still has some shortcomings. TMSP considers the case that mobile devices have one wireless-access-interface. It means TMSP employs reactive handover method. With reactive handover method, mobile devices must terminate the old connection before connecting to a new network¹. This method usually results in long handover latency, which is not acceptable for real-time multimedia applications. Furthermore, TMSP does not specify how to handover across heterogeneous networks. The number of mobile nodes with multiple wireless access interfaces is increasing recently; hence it is desirable to allow these devices to handoff between networks of different wireless technologies to fully make use of them. For example, in office, one would want to use high-speed IEEE 802.11 WLAN network; when one goes out, it is better to handoff to widely available 3G network.

In this thesis, we present the Virtual Handover Management (VHM) [17] [18]. Virtual Handover Management is a *link-layer mobility mechanism*, which supports multiple-wireless-interface mobile devices to achieve seamless handovers across heterogeneous networks. VHM is not a mobility support protocol; rather it is a link-layer handover management extension to these protocols. VHM would be able to integrate into any mobility

---

¹Unless the wireless interface can be associated with more than one access point simultaneously.

CHAPTER 3. VIRTUAL HANDOVER MANAGEMENT IN IPv4

---

support protocol such as TMSP or MIP. In this thesis, we choose to integrate VHM to TMSP, resulting in TMSP-VHM, due to TMPS's advantages over MIP's². VHM supports a number of wireless technologies: Bluetooth, WiFi, WiMAX, GPRS and UMTS. VHM employs the proactive handover method; hence it allows uninterrupted and smooth handovers. VHM adopts the IEEE 802.21 Media Independent Handover for Network Discovery and Triggering Services.

---

²Refer to Section 2.1 for more details

3.1 Proposed Virtual Handover Management

The Virtual Handover Management consists of two components:

- (i) *Virtual Device Driver (VDD)*: is a network device in between *Link layer* and IP layer in the network protocol stack. VDD enables mobile devices to choose which network access interface to use and quickly handoff from one network access interface to another.
- (ii) *Virtual Auto Handover (VAH)*: includes a set of triggering daemons and a decision-making daemon. VAH's triggering daemons continuously monitor the network condition and report *link events* to the decision-making daemon. VAH executes handoffs by commanding VDD to choose the appropriate network access interface to use.

3.1.1 Virtual Device Driver

Virtual Device Driver (VDD) is a special network access interface. Figure 3.1 depicts the position of Virtual Device Driver in the IP/TCP protocol stack. VDD lies below Network (or IP) layer. The upper protocol layers, such as IP, TCP/UDP, see VDD as the only network access interface. VDD lies within Link layer, and above other real network access interfaces such as GPRS/UMTS, WLAN, Bluetooth or WiMAX interfaces. VDD is a virtual network interface; it does not have the transmission and reception capabilities. Instead, VDD relies on the *real* network interfaces for these tasks. At any time, VDD is attached to only one of the real network access interfaces. By attaching VDD to a network interface, the mobile node will be using this interface for network access. Handovers across IP networks, hence, are simply done by detaching VDD from one interface and re-attaching VDD to another. The interface-switching is transparent to upper network layers. Applications in mobile devices are also unaware of which access technologies they are using. Figure 3.2 illustrates the interface-switching done by VDD.

Figure 3.1: Virtual Device Driver in IP/TCPv4 protocol stack.

Figure 3.2: Interface-switching from WiFi interface to 3G interface.

TMSP considers the case where the mobile node has only one access interface. It means the handover involves only one wireless technology, or homogeneous handover. VDD considers the case where mobile devices are equipped with multiple wireless access interfaces. Mobile devices are able to do heterogeneous handover across networks of different technologies, or vertical handover. With only one network interface, TMSP employs reactive handover. On the other hand, VDD employs the proactive handover technique. The proactive handover method is only available when mobile devices are equipped with multiple network access interfaces³. Figure 3.3 illustrates the two handover methods: reactive and proactive. The blurred dashed line marks the moment when the handoff decision is made. In reactive handover, the handover latency is consisted of four components: (1) the movement detection delay, (2) the layer-2, and (3) the layer-3 handover delays as well as (4) TMSP handoff signaling delay. The movement detection is the delay for the mobile node to detect that it has lost the old connection and moved into another network. This delay is illustrated in Figure 3.3 as the connection break right

³or when a network interface is able to associate with two access points at the same time.

CHAPTER 3. VIRTUAL HANDOVER MANAGEMENT IN IPv4

---

before the handoff decision line. The layer-2 handover is the delay for the mobile node to change from one Link layer connection to another. For example, a change of wireless access point is an layer-2 handover. The layer-3 handover is the delay for the mobile node to get a new IP address in the new network, or the CoA address acquisition delay. The layer-2 and layer-3 delays together with the TMSP handoff signaling are presented by the connection break in the mobile node's connection right *after* the handover decision line as seen in Figure 3.3. On the other hand, in proactive handover, before the current connection is broken, the other unused, available network interface would look for and establish a new connection. The handover is performed when the new connection is ready; hence, there is no break in the mobile device's *connection line*. The proactive handover method overlaps the delays caused in the reactive method; therefore, the handover delay is kept minimal and connection is virtually uninterrupted. Additionally, our proactive handover scheme also performs better than the reactive counterpart in terms of packet loss. In the latter, during the handoffs, those packets in transition, before the handoff signaling is completed would be dropped since the mobile node's address has changed. The number of packet lost depends greatly on how fast the handoff decision and the handoff signaling are made. On the contrary, in the proactive approach, packet loss is not a problem. The mobile devices have two network interfaces; after the handoff signaling, the unused network interface is still on and connected, even though the link quality is low. Hence, those packets in transition are still received by the mobile devices. This is achieved due to the anticipated handoff in the VHM which performs handovers before the current connection becomes too weak.

CHAPTER 3. VIRTUAL HANDOVER MANAGEMENT IN IPV4

Figure 3.3: Reactive and Proactive handover methods.

3.1.2 Virtual Auto Handover

Virtual Auto Handover (VAH) is a set of daemon processes. VAH continuously monitors the network condition and performs automatic handoffs when necessary. VAH adopts the Media Independent Handover (MIH) [9] for handover triggering, which is a standard being developed by IEEE 802.21 to enable the handover of IP sessions from one wireless access technology to another, to achieve mobility of end user devices.

Virtual Auto Handover comprises a set of triggering daemons (the sub-daemons) on the wireless interfaces and a central decision-making daemon (the main daemon). Each triggering daemon is in charge of a network interface. The triggering daemons continuously monitor the network condition through the interfaces and fire notifications to the decision-making daemon. There are 3 basic events to be triggered and handled by VAH: (1) Link UP (LU), (2) Link Down (LD), and (3) Link Going Down (LGD). Table.3.1 summarizes these events. Our implementation of the IEEE 802.11 triggering mechanism uses a combination of signal strength and number of retransmissions to generate the events⁴. The decision-making daemon is in charge of the Virtual Device Driver. The main daemon can also send commands down to the triggering daemons to control the network interfaces. Table 3.2 summarizes the command types sent by the main daemon. Upon receiving the triggered events, the main daemon would take the necessary action. For example, on receiving LD event, VAH would prepare for a backup connection and make a handoff decision to avoid connection disruption. On receiving LU event, VAH would compare the current connection with the new one and perform handoff if it wishes to. To handoff from one network to the other, VAH main daemon instructs the VDD to switch to the respective network interface. The decision-making policy is built into the

---

⁴It is expected that the radio measurement criteria will be specified in future standards such as the future IEEE 802.11k. Owing to the lack of specific criteria, we define thresholds for signal strength and number of retransmissions in our implementation.

CHAPTER 3. VIRTUAL HANDOVER MANAGEMENT IN IPV4

automated handover management algorithm, which will be discussed in the subsequent paragraphs.

Table 3.1: Link Event Table.

No	Event Type	Event Name	Remark
1	State Change	Link Up (LU)	L2 connection established
2	State Change	Link Going Down (LGD)	L2 connection broken
3	Predictive	Link Down (LD)	L2 connection going down

Table 3.2: Command Types.

DOWN	:	Bring down the interface
UP	:	WiFi: Bring up the interface and connect to best AP Or connect to the Access Point specified UTMS: Bring up the interface and connect to UTMS network
SCAN	:	WiFi: Scan and report the best Access Point

It is very common nowadays for a mobile device, such as PDA or laptop, to have the following three wireless interfaces: Bluetooth, WiFi and 3G. Therefore, we derive an algorithm, to automate the handover management process, considering the case where a mobile device is equipped with these three wireless interfaces. We denote LU_{WiFi} as the link up event for the WiFi interface, and LU_{3G} as the link up event for the 3G interface, and similarly for other events. The Bluetooth WPAN network is a personal short-range network and is usually set up for ad-hoc purposes. Bluetooth WPAN is most useful for files sharing, Internet connection sharing within Bluetooth-capable devices, or VoIP and personal multimedia entertainment. Therefore, we assume one would rather want to handoff *manually* to a Bluetooth network than automatically. On the other hand, the WiFi and 3G networks are infrastructure networks and are readily available. WiFi network is preferable to 3G network for several reasons. WiFi offers high-speed Internet access with very low cost of service, but within a medium range; whereas 3G is slower in connection and incurs a higher cost of service but offers city-wide ubiquitous coverage.

CHAPTER 3. VIRTUAL HANDOVER MANAGEMENT IN IPV4

Thus the mobile devices might want to use a WiFi connection whenever one is available, and to *fail over* to the 3G connection when the WiFi network is unavailable. We define an algorithm, illustrated in Figure 3.5 and Figure 3.6 to automate the VAH handover management. Figure 3.4 represents the various connection states that a mobile device can get into and the transitions that occur between the states.

As seen in the state diagram of Figure 3.4, the mobile node only handoffs to Bluetooth WPAN at user's will. WiFi is of the highest priority. The mobile node always tries to take advantage of the high-speed WiFi network access whenever possible. Only when the current WiFi connection is weakening and there is no alternative WiFi Access Point, then the mobile node will transfer to the 3G network. It is expected that the 3G network provides complete coverage on populated areas⁵. Therefore, the mobile users always have a secured backup connection to stay connected. However, the mobile node would make a *proactive* handover back to WiFi networks when there is a new one available. As seen from the flowcharts in Figure 3.5 and Figure 3.6, most of the handovers performed are proactive. Proactive handovers are preferred to reactive ones since they are uninterrupted and seamless. There is only one reactive handover when the current WiFi connection is getting worse and an alternative WiFi Access Point is found. Since the mobile device has only one WiFi interface, it has no choice but to drop the current connection and establish a new one. There is an option for making all the handovers proactive by equipping the mobile device with two WiFi interfaces. The mobile node brings up the inactive WiFi interface and establish new connection before breaking the old connection. However, this option might be undesirable due to the power consumption concern.

In Figure 3.5, WiFi interface at the mobile node may detect several available WiFi Access Points. The WiFi interface then chooses the one with highest link quality. The same strategy is applied to Figure 3.6 when the mobile node receives the LGD event

---

⁵This assumption is practical as M1's 3G network offers more than 95% street level coverage in Singapore.

CHAPTER 3. VIRTUAL HANDOVER MANAGEMENT IN IPv4

and looks for an alternative Access Point. The issues with Access Point selection are not addressed in this thesis. Readers who are interested in this issue can refer further to [19], [20].

Figure 3.4: Connection State Diagram of a mobile device with 3 wireless access interfaces: Bluetooth, WiFi and 3G.

VAH not only allows automated handover scheme, but also manual handovers by users. However, the former scheme is the default. This automated scheme requires some triggering services running as daemon processes; hence, it may be not desirable as power consumption should be minimized in mobile devices. In fact, some mobile users are totally aware of which wireless network they should use, and when a network handoff is necessary. In these cases, the manual handovers are more efficient. For example, a PDA user is using a VoIP application, over the high-speed WiFi connection at his office. He then has to go out onto the street. Thus, he has to switch to the lower-speed 3G connection. Therefore, he can deliberately activate the 3G connection and perform a manual handoff at the company hall, before stepping out on the streets. When he goes back to his office, he can switch back to his company’s WiFi network at will. It is clear

CHAPTER 3. VIRTUAL HANDOVER MANAGEMENT IN IPv4

Figure 3.5: Flowchart of the network selection procedure when the mobile node is in 3G network.

that for number of situations like this, mobile users are completely aware of when and to which network one should be using.

Figure 3.6: Flowchart of the network selection procedure when the mobile node is in WiFi network.

3.1.3 VHM Integration into Mobility-Support TMSP

VHM is a handover management scheme; it maintains mobile devices connection and performs network handover with no interruption. VHM only supports mobility at Link layer (layer 2) and Network layer (layer 3). Therefore, VHM needs to be combined with other protocols, such as TMSP or MIP which provides mobility in upper layers: TCP/UDP layer and Application layer, to provide complete mobility support solution. In this project, we choose to integrate VHM into TMSP rather than MIP because of TMSP's simplicity and efficiency. As discussed in Chapter 1, TMSP does not require: (1) the mobile node to have permanent IP address, (2) Home Agent and Foreign Agent; and TMSP can better optimize routing.

In the original TMSP, when the mobile node changes its network, it would use the SIP signaling services to inform the SIP-RS and its correspondent nodes of its new CoA. However, TMSP provides a poor handover management. To detect connection changes, the mobile node checks the status of its current Access Point. The AP status includes its periodic advertisements and link quality. Either when the mobile node receives no advertisements from the AP, or when the mobile node detects that the link quality of the AP drops below a defined threshold after a timeout period, the mobile node would search for an alternative Access Point and then do the necessary signaling. The timeout is necessary to prevent mobile devices from interpreting wrong signals that could lead to false, premature handoff decision. VHM provides a solution for better handover management. VHM monitors the IP connection and performs handovers when necessary. When VHM makes a handoff decision, it triggers TMSP signaling service to inform the parties concerned of the connection changes. TMSP and VHM have been integrated successfully resulting in TMSP-VHM. TMSP-VHM is tested with video streaming (VLC media player [21]) and video conferencing (Ekiga, formerly known as GnomeMeeting [22]) applications. TMSP-VHM achieves a smooth and seamless handover. There is

CHAPTER 3. VIRTUAL HANDOVER MANAGEMENT IN IPv4

---

no observable interruption to video and voice during the handovers. These results are much better than the the original TMSP which normally causes 1 to 3 seconds disruption during handovers. Furthermore, TMSP-VHM supports handovers between heterogeneous networks. Figure 3.3 summarizes the types of handovers supported by the original TMSP and the integrated TMSP-VHM.

Table 3.3: Types of Handovers supported by original TMSP and TMSP-VHM.

Technology	One Interface	Multiple Interfaces
Homogeneous	TMSP	TMSP-VHM
Heterogeneous	Not Applicable	TMSP-VHM

3.2 Virtual Handover Management Implementation

VHM is implemented in Linux kernel 2.6.15. The Linux platform is Ubuntu 6.10 (Edgy Eft) [23]. VDD is implemented as a network device module in the Linux kernel. It is coded in C programming language. In Linux, each network access interface is represented in the kernel as a network device, or network interface. For example, there are Ethernet network devices (eth0, eth1), WiFi network devices (ath0, ath1) or 3G devices (ppp0, ppp1). VDD is also a *virtual* network device. Since, VDD relies on the real interface for packet transmission and reception, let us denote the real network interface as the *slave interface*. When VDD associates itself to an interface, it must:

- (i) impersonate the slave interface by configuring itself with the slave interface's IP address, MAC address, subnet mask.
- (ii) modify the *routing table* so that all the route entries previously go through the slave interface now instead going through the VDD. All the routes associated with other network interfaces are hidden to make upper network layers see VDD as the only network interface.
- (iii) forwards in-coming packets from upper network layers to the slave interface for real transmission. VDD does this by keeping the pointer to the *transmit function* of the slave interface and calling the function when outgoing packets from upper network layers arrive.

When a network interface connects to a new network, it must obtain the IP address and other parameters such as the default gateway, subnet mask, and IP addresses of DNS servers from this network. For example, in 3G network, the IP configuration is done through Point-to-Point Protocol [24]; in Ethernet or 802.11 WLAN, the configuration protocol is Dynamic Host Configuration Protocol (DHCP) [25]. VDD puts a hook in

CHAPTER 3. VIRTUAL HANDOVER MANAGEMENT IN IPV4

---

this IP configuration process to extract and store the IP configuration information for its usage. When the mobile node wants to use a specific network access interface, VDD associates itself with that real interface and then impersonates the interface by self-configuring with the stored configuration information. When an application has outgoing packets to send, it would first look into the routing table and find an interface to forward the packets to. Hence, VDD has to modify the routing table in the system so that all the routing entries previously associated with the real interfaces are now re-associated with VDD. The upper network layer, TCP/UDP, sees VDD as the only network interface in the system. Hence, it always passes outgoing packets to VDD for transmission. VDD plays the role of redirecting packets to the appropriate slave interface for real transmission task. On the contrary, the packet reception does not concern VDD. Incoming packets received by the slave interface are passed directly to the upper layers.

Virtual Auto Handover module is implemented in Python Programming Language [26] with network framework Twisted [27]. VAH is a set of triggering daemons and a main decision-making daemon. Each triggering daemons is in charge of one real network interface. The daemons monitor the current connections and trigger LU, LD or LGD events. Table 3.5 describes a sample Link Up event. As stated previously, a radio measurement criteria will be specified in future standards such as the future IEEE 802.11k [28]. Owing to the lack of the specific criteria, we define link-quality thresholds for these events. We obtain the link-quality of the connection through Link-layer statistic information generated by the Wireless Extensions [29]. Through a number of video-streaming experiments, we come up with the thresholds as shown in Table 3.4. We observe that when link quality is above 30% of the full quality, the video streaming is good with low packet loss rate. Hence, 30% is the threshold for Link Up event. With link quality of 5%, the video received is badly distorted and contains so much noise. Hence the Link Down event is triggered at 5% threshold with a timeout of 3 seconds.

CHAPTER 3. VIRTUAL HANDOVER MANAGEMENT IN IPv4

The timeout is to make sure the link is truly low. When the link quality is dropping continuously for a period for 3 seconds, the daemon triggers the Link Going Down event.

Table 3.4: Triggering Thresholds for Link Events.

Link Up	: Link Quality is more than 30% full quality
Link Down	: Link Quality is less than 5% of full quality for a timeout of 3 seconds
Link Going Down	: Link Quality is dropping in the last 3 seconds

The triggering daemons and the main daemon communicate with each other through the *named pipes*, which is an inter-process communication (IPC) method in Linux. The triggering daemons send the event notifications through separate *named pipes* with the main daemon. The event types are summarized in Table 3.1. The main daemon can send commands down to the triggering daemons. Some of commands may be: bring down the interface, bring up the interface, scan for active networks, and connect to a specific network. Table 3.2 summarizes the commands and their meanings. The main daemon is in user-space, and the VDD is in kernel space; therefore, the main daemon controls the VDD through user-to-kernel *IOCTL*⁶. Table 3.6 describes a sample Command Down.

It is quite simple to integrate VHM into TMSP because TMSP is implemented in Python. Whenever VHM decides a handoff, it also triggers the TMSP mobility signaling service. VHM is a light-weight handover management; with 3170 lines of code, including

⁶IOCTL is short for "Input/output control". Ioctls are typically employed to allow user-space processes to communicate with kernel devices.

Table 3.5: Sample format of a triggering LU event from interface ath0, using NTUWL Access Point.

Interface	: ath0
Access Point	: NTUWL
Technology	: WiFi
Type	: Link Change
Link Status:	: LINK UP

CHAPTER 3. VIRTUAL HANDOVER MANAGEMENT IN IPv4

Table 3.6: Sample format of a command from main daemon to bring down the interface ath0.

Interface	:	ath0
Access Point	:	<i>void</i>
Technology	:	WiFi
Type	:	Command
Command	:	DOWN

the integration part with TMSP. The reason for its simplicity is because VDD is a kernel module, which should be as small as possible so as not to affect the kernel operations; whereas VHA is implemented in Python and is built on available network framework. In the next section, we would study the performance of TMSP-VHM in experiments.

3.3 Experimental Result and Discussion

We study the performance of VHM through experiments. Figure 3.7 shows the experiment layout. The testbed comprises two networks A and B, a SIP-Redirect Server, a correspondent node and a mobile node. The correspondent node is continuously transmitting traffic to the mobile node while the mobile node laptop handoffs across these networks again and again. Depending on which experiments are carried out, Network A and Network B can be WLAN 802.11b or WPAN Bluetooth or 3G. The testbed components are listed in Table 3.7.

Figure 3.7: Experiment Testbed

CHAPTER 3. VIRTUAL HANDOVER MANAGEMENT IN IPV4

Table 3.7: Experiment Components

No.	Device	Hardware Info.	Software Configuration
1	Mobile Node	IBM-T42 laptop Intel-Petium(R)-M processor Billionton Bluetooth dongle Atheros 802.11b wireless cards M1's 3G SIM cards	Ubuntu 6.10 BlueZ MadWifi WvDial [30]
2	Correspondent Node	Dell desktop Intel-Pentium(R) 2.4 GHz processor	Ubuntu 6.10
3	WiFi AP	Linksys WRT54G	OpenWrt
4	Bluetooth AP	Dell desktop Intel-Pentium(R) 2.4 GHz processor with Billionton Bluetooth dongle	Ubuntu 6.10 BlueZ
5	SIP-RS	Dell desktop Intel-Pentium(R) 2.4 GHz processor	Ubuntu 6.10

3.3.1 Reactive versus Proactive Handover Methods

In the section, we compare the performance of the original TMSP's one-interface *reactive handover* method and VHM multiple-interface *proactive handover* method. The testbed layout is illustrated in Figure 3.7. Network A and Network B are WLAN 802.11b networks. WiFi Access Points in these networks are Linksys WRT54G routers. To have more controls over network configurations, we remove the Linksys-proprietary firmware and install on Linux-based OpenWrt firmware [31] instead. The correspondent node and SIP-RS are Intel-Pentium(R) 2.4 GHz processor Dell desktops with the same configuration. The mobile node is a IBM-T42, 1.6 GHz Intel-Pentium(R)-M processor laptop; the mobile node is equipped with Atheros wireless 802.11a cards. In experiments with original TMSP, the mobile node uses only one wireless card; whereas with TMSP integrated with VHM, the mobile node uses two wireless cards. The Atheros wireless cards use MadWifi driver [32]⁷; whereas the Bluetooth Base Station use BlueZ [33] which is the official Linux Bluetooth protocol stack. The mobile node, the correspondent node, Bluetooth Base Station and SIP server run on Ubuntu 6.10 operating system [23].

All network entities, except the mobile node, are connected through a wired network hub. The mobile node is connected to the network through one of the access points in the two networks. In our experiments, the correspondent node and the mobile node are registered with SIP-RS; the correspondent node continuously transmitting traffic to the mobile node while the mobile node moves from one network across the other. We develop our own Python-coded traffic generator. We study VHM performance by measuring the throughput and the handover delay during the handovers.

3.3.1.1 Throughput During The Handovers

In this experiment, the correspondent node transmits UDP traffic to the mobile node at a rate of 768 Kbps with packet size of 700 Bytes per packet. This rate is equal

---

⁷MadWifi is one of the most advanced WLAN drivers available for Linux today.

CHAPTER 3. VIRTUAL HANDOVER MANAGEMENT IN IPV4

to the full motion video and audio for videoconference applications. We measure the throughput received at the mobile node when it moves across networks for a number of times. Figure 3.8 plots the throughput at the mobile node with original TMSP. Figure 3.9 shows the throughput at the mobile node with VHM integrated in TMSP. TMSP provides IP mobility support. It allows on-going sessions in the mobile node to continue during handovers. As seen in Figure 3.8 and Figure 3.9, the throughput curves are continued after the handovers. However, without VHM, the throughput received at the mobile node drops sharply to 0 Kbps during the handovers as shown in Figure 3.8. The distortion in the throughput line is because the proactive method, which is employed by the original TMSP, causes disruption in the connectivity of the mobile node. The throughput is not only affected during the handover; but also after the handover, since it takes some time for the connection to stabilize. The throughput disruption also means that during network handoffs, all packets from the correspondent node to the mobile node are lost.

Figure 3.8: Without VHM: Throughput received at the mobile node when the correspondent node transmits at the rate of 768 Kbps.

CHAPTER 3. VIRTUAL HANDOVER MANAGEMENT IN IPv4

Figure 3.9: With VHM: Throughput received at the mobile node when the correspondent node transmits at the rate of 768 Kbps.

While with VHM integrated in TMSP, in Figure 3.9, the throughput received at the mobile node is not affected at all; it is a smooth curve with no distortion during the handovers. Since VHM employs the proactive handover method, before the old connection is broken, VHM connects the unused network interface to a new network. The mobile node connection is redirected to that interface immediately when it gets ready. Hence, no connection break should be present. VHM’s proactive handover method requires mobile devices to have multiple wireless interfaces. In fact, it is not uncommon nowadays to have multiple wireless interfaces in mobile devices; especially when the price for wireless cards is becoming cheaper. It costs less than \$S20 for a Linksys wireless card. Using two wireless interfaces in mobile devices may seem to pose a problem in terms of power consumption; especially when battery-life is limited. However, since VHM only activates one interface at a time; and the other interface is only brought up during the handover, VHM thus limits battery consumption to a minimum and still enables fast and smooth handovers.

3.3.1.2 Handover Delay

In this experiment, we study the performance of VHM by studying the handover latency. The correspondent node transmits UDP traffic to the mobile node at a rate of 400 Kbps, with packet size of 500 Bytes per packet. The mobile node moves across networks again and again for 1000 times. We capture the packets received at the mobile node. Suppose the mobile node is in Network A and it is moving to Network B, the handover latency is calculated from the arrival time of the last packet from Network A to the arrival time of the first packet from Network B. The experiments are run for 2 cases: when the mobile node uses the original TMSP and when the mobile node uses the VHM-TMSP. We then calculate the *cumulative distribution function* (CDF) of the handover latencies and also the interarrival times.

Figure 3.10: CDF of handover latencies with and without VHM integrated in TMSP

Figure 3.10 shows the 3 CDF plots: handover delays with and without VHM and the packet interarrival. We plot the packet interarrival times CDF for comparison with handover delays. From Figure 3.10, we observe that the handover delays when using

CHAPTER 3. VIRTUAL HANDOVER MANAGEMENT IN IPv4

VHM is almost the same as the packet interarrival. This is expected as the proactive handover method virtually introduces no break in connectivity. The original TMSP, with reactive method, causes longer handover delays. The average packet interarrival is 12.159 milliseconds. VHM average handover delays is 12.317 milliseconds, while no-VHM average handover delay is 1.503 seconds. As the CDF of no-VHM case is too big compared to CDF of VHM case, we plot a smaller image of the CDFs of handover delays with VHM and packet interarrival inside Figure 3.10 for a clearer view.

3.3.2 Vertical Handover across Heterogeneous Networks

In this section, we study the performance of TMSP-VHM during vertical handovers. We did experiments with three wireless technologies: WLAN 802.11b, Bluetooth and 3G⁸. We set two testbeds. The first testbed is Bluetooth-WiFi; its layout is illustrated in Figure 3.7. The second testbed is 3G-WiFi; the layout is described in Figure 3.11. The former testbed is set in CeMNet lab; whereas the latter uses the real network infrastructure. We choose to test WiFi against Bluetooth and 3G because WiFi is the most prominent wireless technology nowadays. It is desirable to handoff between a personal-area-network (Bluetooth WPAN) or a wide-area-network (3G) and famous local area network (WLAN).

For the Bluetooth-WiFi testbed in Figure 3.7, Network A is a 802.11b WLAN; and Network B is a Bluetooth WPAN. The Access Point in Network A is a Linksys WRT54G router; whereas Bluetooth Base Station is a Intel-Pentium(R) 2.4 GHz processor Dell desktop with a Billionton Bluetooth dongle. The correspondent node and SIP-RS are also Dell desktops with the same configuration. The mobile node is a IBM-T42, Intel-Pentium(R)-M processor laptop; the mobile node is equipped with a Atheros wireless 802.11a card and a Billionton Bluetooth dongle. All network entities, except the mobile node, are connected through a wired network hub. The mobile node are connected to the network through one of the access points in the two networks.

For the 3G-WiFi testbed in Figure 3.11, we make use of NTU Wireless as the WiFi network; and the 3G/W-CDMA network is set by M1. For ease of setup, the mobile node, the correspondent node and SIP-RS are all placed inside NTU network. The mobile node continuously moves back and forth between the two networks.

In our experiments, the correspondent node and the mobile node register themselves with SIP-RS; the correspondent node continuously transmitting traffic to the mobile node while the mobile node moves from one network across the other. We study the

---

⁸We use the 3G network setup in Singapore by M1.

performance of VHM by measuring the throughput and the handover delay during the handovers.

Figure 3.11: 3G-WiFi testbed layout

3.3.2.1 Throughput During The Handovers

Here, we study the performance of TMSP-VHM by measuring the throughput during the handovers. The correspondent node transmits UDP traffic to the mobile node while the mobile node handoffs back and forth between WiFi network and Bluetooth network (or 3G network). We run the experiment twice for each testbed. For the Bluetooth-WiFi testbed, in the first run, the correspondent node sends the videoconference traffic at a rate of 768 Kbps with packet size of 700 Bytes per packet. This sending rate is lower than either WiFi bandwidth (11 Mbps) or Bluetooth bandwidth (1.6 Mbps). In the second run, the correspondent node sends the traffic at a rate of 4 Mbps. This rate is lower than WiFi bandwidth but higher than Bluetooth bandwidth. We measure the throughput received at the mobile node when the mobile node moves back and forth across the networks every 100 seconds.

For the Bluetooth-WiFi testbed, Figure 3.12 and Figure 3.13 plot the throughput at the mobile node with the correspondent node sending at rates of 768 Kbps and 4 Mbps respectively. In Figure 3.12 and Figure 3.13, the mobile node handoffs back and

CHAPTER 3. VIRTUAL HANDOVER MANAGEMENT IN IPv4

Figure 3.12: Bluetooth-WiFi: Throughput received at the mobile node when the correspondent node transmits at the rate of 768 Kbps.

Figure 3.13: Bluetooth-WiFi: Throughput received at the mobile node when the correspondent node transmits at a high rate of 4Mbps.

CHAPTER 3. VIRTUAL HANDOVER MANAGEMENT IN IPV4

Figure 3.14: 3G-WiFi: Throughput received at the mobile node when the correspondent node transmits at half of the rate of 768 Kbps.

Figure 3.15: 3G-WiFi: Throughput received at the mobile node when the correspondent node transmits traffic at a rate of 1 Mbps.

CHAPTER 3. VIRTUAL HANDOVER MANAGEMENT IN IPV4

---

forth 9 times between 802.11b WLAN and Bluetooth WPAN. From the two figures, the throughputs are continued after the handovers; it again shows that TMSP successfully supports IP mobility. On the other hand, VHM allows the mobile node to handoff smoothly and seamlessly across networks as the throughput does not drop to zero during the handover. When the correspondent node is sending at a rate of 768 Kbps, as seen in Figure 3.12, the throughput is almost a straight line since this rate is lower than the bandwidth capacity of both networks. When the mobile node is connected to Bluetooth WPAN, the throughput line, however, is jerkier with a lot of spikes; while when the mobile node is connected to 802.11b WLAN, the throughput is smoother. It is because the videoconference rate of 768 Kbps is far less than the 802.11b WLAN bandwidth of 11 Mbps; while it is already half of the bandwidth of Bluetooth WPAN of 1.6 Mbps. The jerky throughput line also implies that the packet interarrival jitter (the end-to-end delay variation) is bigger. There is a need to keep jitter to a minimum for real time multimedia applications as jitter affects the play-out delay at the receiver.

When the correspondent node is sending at a rate of 4 Mbps in Figure 3.13, the throughput goes up and down after each handover. It is because the sending rate of 4 Mbps is higher than the Bluetooth bandwidth and lower than the WLAN 802.11b's. When the mobile node is at the Bluetooth WPAN, the network can afford only a maximum rate of 1.6 Mbps. The rest of the traffic, which is 4 Mbps minus 1.6 Mbps i.e. 2.4 Mbps, is dropped. When the mobile node switches back to 802.11b WLAN network, there is a soar in the throughput curve. It is because the mobile node still receives the un-forwarded and buffered traffic from the Bluetooth Base Station. Note that, during the experiments, the two access interfaces are always on. Similar to Figure 3.12, when the mobile node is at the Bluetooth WPAN network, the throughput line is jerkier than when it is at the 802.11b WLAN. Different wireless technologies possess different ranges and speeds. It is good to take advantage of all of them but there should be a QoS scheme

CHAPTER 3. VIRTUAL HANDOVER MANAGEMENT IN IPv4

---

implemented to help the real-time multimedia applications to better match the access capabilities. We discuss this issue and our solution to the problem in Chapter 5.

For the 3G-WiFi testbed, Figure 3.14 and Figure 3.15 plot the throughput at the mobile node with the correspondent node sending at rates of 384 Kbps and 1 Mbps respectively. The rate of 384 Kbps is lower than either WiFi bandwidth (11 Mbps) or 3G bandwidth (500 Kbps). Whereas the rate of 1 Mbps is lower than WiFi bandwidth but higher than 3G bandwidth. The same observation is noted and can be explained with the same explanation as the Bluetooth-WiFi case.

3.3.2.2 Handover Delay

In this experiment, we study the performance of VHM by studying the delay of the handovers. The correspondent node transmits UDP traffic to the mobile node while the mobile node moves across 802.11b WLAN and Bluetooth WPAN network (or 3G network) back and forth for 1000 times. We run the experiments three times for each testbed (Bluetooth-WiFi and 3G-WiFi): the first run, the correspondent node transmits at the telephone quality rate of 8 Kbps with a packet size of 140 Bytes. In the second run, the correspondent node transmits at the videophone quality rate of 64 Kbps with a packet size of 700 Bytes; and the last run uses a high rate of 400 Kbps with a packet size of 500 Bytes. We capture the packets received at the mobile node. The handover latency is calculated from the arrival time of the last packet from the former network to the arrival time of the first packet in the new network. We categorize two types of handover delays: from Bluetooth (or 3G) to WiFi and from WiFi to Bluetooth (or 3G). We also compare the handover delays with the packet interarrival as they are inter-related. We then calculate the cumulative distribution function (CDF) of the handover latencies and the packet interarrival times.

Figure 3.16, Figure 3.17 and Figure 3.18 show the plots for the three experimental runs in Bluetooth-WiFi testbed; whereas Figure 3.19, Figure 3.20 and Figure 3.21 are from the 3G-WiFi testbed. With 3G-WiFi *real* testbed, the plots are rough and not as smooth as these from in-lab Bluetooth-WiFi testbed; and the 3G-WiFi packet interarrival and handover delays are more varied with bigger variation. Nevertheless, they imply the same results. As observed from the six figures, the handover delays are small and approximately close to the packet interarrival time. It means VHM performs fast and seamless vertical handovers across heterogeneous networks. With Virtual Device Driver in the kernel, VHM is able to do fast vertical handovers by just switching VDD attachment from one access interface to another. VDD is tested and confirmed to support many currently

CHAPTER 3. VIRTUAL HANDOVER MANAGEMENT IN IPV4

Figure 3.16: Bluetooth-WiFi: CDF of handover delays of vertical handovers when the correspondent node transmits at the rate of 8 Kbps.

Figure 3.17: Bluetooth-WiFi: CDF of handover delays of vertical handovers when the correspondent node transmits at the rate of 64 Kbps.

CHAPTER 3. VIRTUAL HANDOVER MANAGEMENT IN IPv4

Figure 3.18: Bluetooth-WiFi: CDF of handover delays of vertical handovers when the correspondent node transmits at the rate of 400 Kbps.

Figure 3.19: 3G-WiFi: CDF of handover delays of vertical handovers when the correspondent node transmits at the rate of 8 Kbps.

CHAPTER 3. VIRTUAL HANDOVER MANAGEMENT IN IPv4

Figure 3.20: 3G-WiFi: CDF of handover delays of vertical handovers when the correspondent node transmits at the rate of 64 Kbps.

Figure 3.21: 3G-WiFi: CDF of handover delay of vertical handovers when the correspondent node transmits at the rate of 400 Kbps.

CHAPTER 3. VIRTUAL HANDOVER MANAGEMENT IN IPV4

emerging wireless access technologies: Bluetooth, WiFi, WiMAX, GPRS, UMTS and HSDPA.

Figure 3.22: Illustration of handoff from Bluetooth and WiFi and vice versa.

One common characteristic from the CDF plots in the six figures is that handovers from Bluetooth (or 3G) to WiFi takes less time than handovers from WiFi to Bluetooth (or 3G); and the interarrival time is in between the two, i.e. less than handoff delays from Bluetooth (or 3G) to WiFi and more than the handoff delays from WiFi to Bluetooth (or 3G). This is explained in Figure 3.22, which illustrates the situation during vertical handovers from Bluetooth to WiFi and vice versa in the time axis. Since packets are sent at equal intervals, so the inter-departure time at the correspondent node is illustrated as fix solid lines in Figure 3.22. The interarrival time is approximately equal to the inter-departure time with small variance due to network factors such as: transmission time, queuing delay in routers. The packet end-to-end delay in Bluetooth network is illustrated as blue long-dashed lines; while the packet end-to-end delay in WiFi network

CHAPTER 3. VIRTUAL HANDOVER MANAGEMENT IN IPv4

---

is illustrated as red short-dashed lines. The lower speed (compared to WiFi) and the packet buffering of the Bluetooth network (or 3G network) cause longer end-to-end delay, which are depicted as longer lines. Figure 3.22 illustrates the scenario just before and right after the handoff moment, when the last packet from the old network and the first packet from the new network are sent. The handover delay is then calculated as from the arrival time of the last packet to the arrival time of the first packet. Hence, handover delays are depicted as the arrow-head lines by connecting the arrival time points of the last packet and the first packet. As observed in Figure 3.22, the Bluetooth-to-WiFi handover delay line is shorter than the WiFi-to-Bluetooth handover line. And the inter-departure time line, which is approximately equal to the interarrival time, is longer than the former and shorter than the latter. This is in line with our experiment observations in the earlier CDF figures. The same explanation is applied to the 3G-WiFi case.

In general, VHM performs well in managing the vertical handovers across heterogeneous networks. The handover latency is minimal and close to the packet interarrival time. VHM enables mobile devices to handoff smoothly and without disruption across either homogeneous or heterogeneous networks. VHM is essential to real-time multimedia applications where seamless handoff is strongly desirable.

3.4 Summary

In this chapter, we present a link-layer mobility scheme: *Virtual Handover Management*. Virtual Handover Management provides mobility support at the Link layer, which allows multiple-access-interface mobile devices to achieve fast and efficient handoffs across heterogeneous networks. VHM supports a number of emerging wireless technologies: Bluetooth, WiFi, WiMAX, GPRS and UMTS. VHM employs the proactive handover method; hence it allows uninterrupted and smooth handoffs. VHM serves as an extension to the existing mobility support protocols such as Mobile IP and TMSP. In this thesis, VHM is integrated into TMSP, resulting in TMSP-VHM. The integrated TMSP-VHM is a complete mobility solution to the current heterogeneous networks. TMSP-VHM is tested successfully in the real network environment. TMSP-VHM is proved to provide smooth and uninterrupted experience for mobile devices when moving across networks of different technologies.

Chapter 4

Virtual Handover Management in IPv6

TMSPv6 is designed for mobility support in IPv6. Like TMSP, TMSPv6 shares some shortcomings. Firstly, TMSPv6 employs the reactive handover method. A mobile node must terminate the old connection before being able to establish a new one. This method usually results in long handover latency and high packet loss, which are not desirable for real-time multimedia applications. Furthermore, TMSPv6 does not attempt to solve the vertical handover problem, which requires an efficient link-layer management scheme.

In this chapter, we extend the Virtual Handover Management solution to IPv6 (VHMv6). VHMv6 is a *link-layer mobility mechanism* to support fast transition across heterogeneous networks. VHMv6 is an extension to IP mobility support protocols. Like the IPv4 case, we choose to integrate VHMv6 to TMSPv6 (TMSP-VHMv6) due to TMSPv6's advantages over MIPv6's. VHMv6 supports a number of wireless technologies: Bluetooth, WiFi, WiMAX, GPRS and UMTS. VHMv6 employs proactive handover method which allows uninterrupted and smooth handovers. VHM follows the IEEE 802.21 Media Independent Handover for Network Discovery and Triggering Services. The VHMv6 mechanism shares the same idea with VHMv4. The details of VHM operation is presented in Section 3.1. We only discuss the VHMv6's implementation in the following section.

4.1 Virtual Handover Management Implementation

VHMv6 is implemented in Linux kernel 2.6.15. The Linux platform is Ubuntu 6.10 (Edgy Eft) [23]. In this section, we do not go into the detailed implementation of VHMv6 since VHMv4 and VHMv6 shares the same working mechanism. We rather present the extra implementation work needed to extend VHMv4 into VHMv6.

Virtual Auto Handover IPv6 module is implemented in Python Programming Language [26] with the Twisted network framework [27]. VAHv6 is a set of triggering daemons and a main decision-making daemon. The VAHv6 module is reused from the VAHv4 and no extra implementation work is needed.

VDDv6 is implemented as a network device module in the Linux kernel, and is written in C programming language. Since, VDDv6 relies on the slave interface for packet transmission and reception, when VDDv6 associates itself to an interface, it must first impersonate the slave interface by self-configuring with the slave interface's IP address, MAC address, subnet mask. Secondly, VDDv6 must modify the *system routing table* so that all the route entries associated with the slave interface now go through the VDD. The routes associated with other network interfaces are hidden to make upper network layers see VDDv6 as the only network interface. Thirdly, VDDv6 keeps the pointer to the *transmit function* of the slave interface and calls it when in-coming packets arrive. By doing this, VDDv6 can rely on the slave interface for real transmission.

When a network interface connects to a new network, it must obtain the IP address and other parameters such as the default gateway, subnet mask, and IP addresses of DNS servers from this network. A highly useful aspect of IPv6 [34] is its ability to automatically configure itself without a stateful configuration protocol, such as *Dynamic Host Configuration Protocol* for IPv6 (DHCPv6). The *IPv6 Address autoconfiguration* is described in [35]. By default, an IPv6 node can configure a link-local IP address for each interface. By using *Router Discovery*, a host can also determine the addresses

CHAPTER 4. VIRTUAL HANDOVER MANAGEMENT IN IPv6

---

of routers, additional IP addresses, and other configuration parameters. The IP self-configuration is then done automatically in kernel. IPv6 nodes use *Neighbor Discovery* [36] to exchange configuration messages with routers. When VDDv6 is attached to a network interface, it must self-configure to impersonate that interface. To achieve that, we deploy a *virtual hook* in the Neighbor Discovery subsystem. Upon receiving a Router Discovery packet, Neighbor Discovery not only configures the slave interface, but also configures the VDDv6.

When a node has a packet to send to its neighbor, either the gateway router or the local nodes, but does not know the neighbor's link-layer address, it performs the address resolution. In IPv4, the address resolution relies on Address Resolution Protocol (ARP) [37]. In order to do this, ARP sends an ARP request packet to all the local nodes using a broadcast address. The broadcasting request packet is sent out by the access interface which receives the packets from the upper network layers. All local nodes will receive this request packet. If a node receives the ARP request and it is the destination, it will respond with the link-layer address. When VDDv4 receives outgoing packets from the upper network layers, it also has to perform link-layer address resolution. Since VDDv4 has no packet receiving mechanism, VDDv4 has no way to receive the link-layer address reply from the neighbor. Hence, the address resolution is not able to be completed. Fortunately, ARP supports *old devices* that have not been adapted to the new neighboring infrastructure. It means that VDDv4 can still do the address resolution by relying on slave device.

In IPv6, the link-layer address resolution is performed by Neighbour Discovery (ND) protocol [36]. ND protocol is a new protocol and unlike its IPv4 counterpart ARP, ND protocol does not support one interface to rely on other interface for the link-layer address resolution. ND works in much the same way as APR with a slight difference. To do the address resolution, a network interface creates a *Neighbor Cache* entry in the

CHAPTER 4. VIRTUAL HANDOVER MANAGEMENT IN IPV6

INCOMPLETE state and broadcasts a *Neighbor Solicitation* message targeted at the destination neighbor. The destination node then replies with a *Neighbor Advertisement* in response to a valid Neighbor Solicitation. At the mobile node, when a valid Neighbor Advertisement is received, the network interface which initiated the address resolution would take up the necessary steps to finish it. The Neighbor Cache is searched for the target's entry and ND marks the entry as *COMPLETE*. The outgoing packets that wait for the address resolution to complete are now sent out. The other network interfaces, which are not interested in the Neighbor Advertisement reply, simply discard it. When VDDv6 receives outgoing packets from the upper network layers for transmission, it has to perform the link-layer address resolution too. VDDv6 can rely on the slave interface for transmission of the Neighbor Solicitation message. However, like VDDv4, VDDv6 does not have the reception capability. The Neighbor Advertisement replied by the destination node is always received by the slave interface. Since the slave interface is not the one which initiated the address resolution, it simply discards the reply message by default. That way, VDDv6 would always miss the advertisement and can never resolve the link-layer address. We get over this address resolution problem by extending the *virtual hook* in Neighbor Discovery sub-system. The virtual hook prevents the slave interface from discarding the reply advertisement but instead pass the message to VDDv6. Hence, the address resolution initiated by VDDv6 can complete successfully.

The Virtual Handover Management IPv6 is extended from the Virtual Handover Management IPv4. For the ease of explanation, we separate the discussion of VHA_{v6} and VHA_{v4} in this thesis. However, the current integrated VHA version works with both IPv4 and IPv6 simultaneously.

4.2 Experimental Result and Discussion

We study the performance of TMSP-VHMv6 through experiments. We use the same testbed as with IPv4, which is shown in Figure 3.7. The testbed components are listed in Table 3.7, Section 3.3. The testbed comprises two networks A and B, a SIP-Redirect Server, a correspondent node and a mobile node. The correspondent node is continuously transmitting traffic to the mobile node while the mobile node laptop handoffs across the two networks again and again. Depending on which experiments are carried out, Network A and Network B can be WLAN 802.11b or WPAN Bluetooth. The experiment results and plots are very similar to the VHMv4. It is understandable because VHMv4 and VHMv6 share the same mechanism and handover methods. The only difference is the underlying routing protocol, IPv4 and IPv6. In fact, IPv4 and IPv6 also differ only in the network address scheme; and they share similar routing mechanism. Hence, in this experiment section, we only present the results and plots. Explanations which are similar to those in Chapter 3 would not be repeated.

4.2.1 Reactive versus Proactive Handover Methods

Here, we compare the performance of the original TMSPv6 one-interface *reactive handover* method and VHMv6 multiple-interface *proactive handover* method. The testbed layout is illustrated in Figure 3.7. Network A and Network B are both WLAN 802.11b networks. WiFi Access Points in these networks are Linksys WRT54G routers. The correspondent node and SIP-RS are Intel-Pentium(R) 2.4 GHz processor Dell desktops with the same configuration. The mobile node is a IBM-T42, Intel-Pentium(R)-M processor laptop; the mobile node is equipped with an Atheros wireless 802.11a cards. In experiments with original TMSPv6, the mobile node has only one wireless card; whereas in TMSPv6 integrated with VHMv6, the mobile node uses two wireless cards. The mobile node is connected to the network through one of the access points. In our experiments,

CHAPTER 4. VIRTUAL HANDOVER MANAGEMENT IN IPV6

---

the correspondent node and the mobile node are registered with SIP-RS; the correspondent node continuously transmitting traffic to the mobile node while the mobile node moves from one network across the other. The throughput and the handoff delay are measured and plotted.

4.2.1.1 Throughput During The Handovers

In this experiment, the correspondent node transmits UDP traffic to the mobile node at a rate of 768 Kbps with packet size of 700 Bytes per packet. This rate is equal to the full motion video audio rate for videoconference applications. The throughput received at the mobile node is measured when it moves across networks for a number of times.

Figure 4.1 plots the throughput at the mobile node with original TMSPv6. Figure 4.2 shows the throughput at the mobile node with TMSP-VHMv6. TMSPv6 provides IP mobility support so that the on-going sessions in the mobile node can continue when the mobile node handoffs. In Figure 4.1, without VHM, the throughput received at the mobile node drops sharply to 0 Kbps during the handovers. The original TMSPv6's reactive method causes a disruption in the connectivity of the mobile node. The throughput disruption also implies that during handover, all packets from the correspondent node to the mobile node are lost. On the other hand, VHMv6 employs the proactive handover method. As seen in Figure 4.2, with VHMv6 integrated in TMSPv6, the throughput received at the mobile node is a smooth line with no distortion during the handovers.

CHAPTER 4. VIRTUAL HANDOVER MANAGEMENT IN IPV6

Figure 4.1: Without VHMv6: Throughput received at the mobile node when the correspondent node transmits at the rate of 768 Kbps.

Figure 4.2: With VHMv6: Throughput received at the mobile node when the correspondent node transmits at the rate of 768 Kbps.

4.2.1.2 Handover Delay

In this experiment, we measure the handover latency by VHMv6. The correspondent node transmits UDP traffic to the mobile node at a rate of 400 Kbps with packet size of 500 Bytes per packet. The mobile node moves across networks again and again for 1000 times. The experiments are run for 2 cases: when the mobile node uses the original TMSPv6 and when the mobile node uses the VHM-TMSPv6. We then calculate the *cumulative distribution function* (CDF) of the handover latencies and also the interarrival times.

Figure 4.3: CDF of handover delays with and without VHMv6.

Figure 4.3 shows the 3 CDF plots: handover delays with and without VHM, and the packet interarrival time. VHMv6’s proactive handover method causes delays almost equal to the packet interarrival time. TMSPv6 reactive method, however, causes much longer handover delays, with an average of more than 1 second.

4.2.2 Vertical Handover across Heterogeneous Networks

In the section, we study the performance of TMSP-VHMv6 during the vertical handovers. We tested with two wireless technologies: WLAN 802.11b, Bluetooth. Since there is currently no available 3G IPv6 network, we were unable to experiment TMSP-VHMv6 with the 3G-WiFi testbed in IPv6 as in IPv4 case. The Bluetooth-WiFi testbed layout is illustrated in Figure 3.7. Network A is a 802.11b WLAN; and Network B is a Bluetooth WPAN. The testbed components are listed in Table 3.7. We study the performance of VHMv6 by measuring the throughput, the handover delay during the handovers.

4.2.2.1 Throughput During The Handovers

We study the performance of TMSP-VHMv6 by measuring the received throughput at the mobile node during the handovers. The correspondent node transmits UDP traffic to the mobile node while the mobile node handoffs back and forth between WiFi network and Bluetooth network. We made two experiments runs. The first run, the correspondent node sends the videoconference traffic at a rate of 768 Kbps with packet size of 700 Bytes per packet. In the second run, the correspondent node sends the traffic at at rate of 4 Mbps.

Figure 4.4 and Figure 4.5 show the throughput at the mobile node with the correspondent node sending at rates of 768 Kbps and 4 Mbps respectively. While TMSPv6 enables session continuity during network handovers, VHMv6 allows the mobile node to handoff seamlessly and without interruption across heterogeneous networks. When the correspondent node is sending at a rate of 768 Kbps, as seen in Figure 4.4, the throughput is almost a straight line; because this rate is lower than the bandwidth capacities of both networks. When the correspondent node is sending at a rate of 4 Mbps, as seen in Figure 4.5, the throughput line is up and down since this rate is higher than the Bluetooth bandwidth of 1.6 Mbps, and lower than the WiFi bandwidth of 11 Mbps. When

CHAPTER 4. VIRTUAL HANDOVER MANAGEMENT IN IPV6

Figure 4.4: Bluetooth-WiFi IPv6: Throughput received at the mobile node when the correspondent node transmits at the rate of 768 Kbps.

Figure 4.5: Bluetooth-WiFi IPv6: Throughput received at the mobile node when the correspondent node transmits at a high rate of 4 Mbps.

CHAPTER 4. VIRTUAL HANDOVER MANAGEMENT IN IPV6

---

the mobile node is at the Bluetooth network, the difference between the sending rate and the Bluetooth bandwidth is dropped either by the Bluetooth Base Station.

4.2.2.2 Handover Delay

This section studies the performance of VHMv6 by studying the delay during vertical handovers. We made three experiment runs. In the first run, the correspondent node transmits at the telephone quality rate of 8 Kbps with packet size of 140 Bytes per packet. In the second run, the correspondent node transmits at the videophone quality rate of 64 Kbps with packet size of 700 Bytes per packet; and for the last run, the correspondent node transmits at a high rate of 400 Kbps with 500 Bytes per packet. Handover delays are categorized into two delays: from Bluetooth to WiFi and from WiFi to Bluetooth. We then calculate the cumulative distribution functions of the handover latencies and also the packet interarrival times for comparison purpose.

Figure 4.6, Figure 4.7 and Figure 4.8 show the plots for the three experiment runs. As observed from the three figures, the handover delays are small and approximately close to the packet interarrival. It means VHMv6 performs fast and seamless vertical handovers across heterogeneous networks. One common characteristic from the CDF plots in the three figures of Figure 4.6, Figure 4.7 and Figure 4.8 is that handovers from Bluetooth to WiFi takes less time than handovers from WiFi to Bluetooth (or 3G); and the interarrival time is in-between the two, i.e. less than handoff delay from Bluetooth (or 3G) to WiFi and more than the handoff delay from WiFi to Bluetooth (or 3G). The detailed explanation is in Section 3.3.2.2. In general, VHMv6 performs well in managing the vertical handovers across heterogeneous networks. The handover latency is kept to a minimum and close to the packet interarrival time. VHMv6 enables mobile devices to experience fast and uninterrupted transition across heterogeneous networks. VHMv6 is essential to real-time multimedia applications where seamless handoff is strongly desirable.

CHAPTER 4. VIRTUAL HANDOVER MANAGEMENT IN IPV6

Figure 4.6: Bluetooth-WiFi IPv6: CDF of handover delay of vertical handovers when the correspondent node transmits at the rate of 8 Kbps.

Figure 4.7: Bluetooth-WiFi IPv6: CDF of handover delay of vertical handovers when the correspondent node transmits at the rate of 64 Kbps.

Figure 4.8: Bluetooth-WiFi IPv6: CDF of handover delay of vertical handovers when the correspondent node transmits at the rate of 400 Kbps.

4.3 Summary

In this chapter, we extend the *Virtual Handover Management* solution to IPv6, namely VHMv6. VHMv6 is a link-layer mobility mechanism to support uninterrupted transition across networks of heterogeneous technologies. VHMv6 supports a number of wireless technologies: Bluetooth, WiFi, WiMAX, GPRS and UMTS. VHMv6 employs proactive handover method which allows uninterrupted and smooth network handoffs. VHMv6 serves an extension to the existing IP mobility support protocols such as TMSPv6 and MIPv6. Like the IPv4 case, VHMv6 is integrated with TMSPv6, resulting in TMSP-VHMv6. The integrated TMSP-VHMv6 provides a complete mobility solution for heterogeneous wireless networks. TMSP-VHMv6 is implemented in Linux and in Python programming language. TMSP-IPv6 is tested and proved to work efficiently.

Chapter 5

Quality of Service in Handover Management

In this part of the thesis, we propose a QoS framework [38] for QoS provisioning in the mobility scenario. The QoS framework caters to QoS-aware applications so that they can adapt to the variability of bandwidth and delay across heterogeneous wireless technologies. Our QoS framework allows QoS-aware applications to transit *seamlessly* and *smoothly* across heterogeneous networks.

5.1 Proposed QoS Framework in TMSP-VHM

We develop a QoS framework in TMSP-VHM to enable adaptive applications to dynamically change their attributes according to the capability of the current connection. Owing to the variability of the wireless access technologies, dynamic adaptation is necessary to maintain users' perceived QoS. For example, when a user moves from a high-speed WiFi connection to a UMTS network, applications' adaptation of the bit rates is necessary to fit in the new lowered bandwidth. This adaptation can be done, for example, through transcoding, using different video encoding rate, or enabling and disabling high-quality video streams. Our QoS framework reduces the signaling cost to a minimum because it combines the QoS signaling with handover signaling.

CHAPTER 5. QUALITY OF SERVICE IN HANDOVER MANAGEMENT

We do not differentiate horizontal and vertical mobility. We propose a generic QoS framework which works seamlessly across both horizontal and vertical mobility. Our scheme applies to any handovers in IP-based networks. In this section, we first describe the *end-to-end application signaling* through SIP. Then we describe the *network QoS signaling for end-to-end reservation* in TMSP-VHM.

5.1.1 End-to-end SIP Application Signaling

In our design, we choose to extend SIP signaling for QoS. We introduce new SIP messages for QoS purposes: *QueryQoS* and *SetQoS*. Using these messages, mobile devices inform each other about the current access capabilities. The SIP server is also extended to be able to handle these QoS messages for initiation purpose. We assume that the QoS signaling traffic is transmitted through best effort class in the networks. This is to cater for scenarios where QoS support does not exist, which is still common in the current *last mile* wireless access network. As the SIP application signaling can be made independent of the network QoS signaling, we separate the designs for application and network QoS signaling.

When a mobile node retrieves the IP address of the correspondent node using the correspondent node's SIP URI, the mobile node will also retrieve the correspondent node's QoS setting from the SIP server using *QueryQoS* message. The QoS setting in the SIP server is updated by the mobile nodes when their connections change. Here we assume that the mobile nodes have the capability to know the current QoS capability of its connection. The mobile nodes can perform probing or analysis on their connections to establish the QoS (e.g. [39, 40]). The mobile node will also inform the correspondent node about its current QoS capabilities using *SetQoS* message. Having this QoS information, applications can start with appropriate parameters that fit the current QoS capabilities of both mobile nodes and correspondent nodes.

There are two scenarios that we need to consider which are proactive handover and reactive handover. First, let us consider the proactive handover scenario. In proactive handover, an mobile node establishes a new connection through the new available interface before it breaks the old connection. Figure 5.1 illustrates the proactive handover procedure with the QoS signaling. In this scenario, the QoS signaling can be done in the old or the new connection. Here we specify that if the new QoS is a subset of (or

CHAPTER 5. QUALITY OF SERVICE IN HANDOVER MANAGEMENT

Figure 5.1: Description of session handover protocol and QoS signaling in proactive handover.

Figure 5.2: Description of session handover protocol and QoS signaling in reactive handover.

CHAPTER 5. QUALITY OF SERVICE IN HANDOVER MANAGEMENT

---

lesser than) the old QoS then the QoS signaling is done in the old connection *before* the transfer to the new connection. If the new QoS is a superset of (or greater than) the old QoS or the comparison is not strictly greater or lesser, the QoS signaling is done in the new connection *after* the transfer of connection to the new interface. After the new QoS is established, applications can adjust their parameters to suit the new connection's QoS capabilities. Using this QoS framework, QoS updates are seamless across handovers.

In reactive handovers, when the mobile node's connection breaks, new connection is established through the new interface; the QoS signaling is done after the new connection is established. Figure 5.2 illustrates the reactive handover procedure with the QoS signaling. Similar to proactive handover, before the QoS signaling, the mobile node needs to establish the new connection QoS capabilities. After establishing a new connection and updating the correspondent node about the new QoS capabilities, both the mobile node and the correspondent node can update its applications to suit the new QoS capabilities.

5.1.2 End-to-end Network QoS Signaling

For network QoS signaling and reservation, we adopt the resource reservation architecture introduced by Hillebrand *et. al.* [16]. Depending on the network nodes capabilities and support, other architecture, such as [14]¹, can be easily integrated with ours.

We further assume a centralized QoS architecture where a *Domain Resource Manager* (DRM) exists for each domain. DRM is responsible for QoS management in a network domain. Following the design in [16], DRM will determine the end-to-end path from the mobile node to the correspondent node and will perform reservation with DRMs for each domain in the path. DRM manages a domain that contains multiple *access routers* (ARs). AR acts as the gateway for wireless access points which provide wireless connectivity to the terminals. Considering this infrastructure in place, when an mobile node establishes a new QoS connection, it registers with the DRM of its domain to establish an end-to-end reservation from the mobile node's domain to the correspondent node's domain. Figure 5.3 shows the illustration of this setup. A possible handover scenario is shown in the figure. This infrastructure setup is compatible with both DiffServ and IntServ.

We further assume that the wireless access points can support QoS reservation and provisioning. Thus, terminals have end-to-end QoS support. Having QoS support from wireless access point also means that we have accurate QoS profile for the SIP application signaling instead of having to perform probing or analysis.

Similar to application signaling, there are two scenarios to consider: proactive handover and reactive handover. In proactive handover, two options are available for QoS signaling: (1) the mobile node informs the current DRM to transfer the reservation to the new DRM; and (2) the mobile node establishes a new reservation to the new DRM and terminates the current reservation. Option 2 is less complex, while option 1 allows faster reservation if the end-to-end paths overlap. Option 1 is specified to be the default

---

¹See also [15] for other approaches.

CHAPTER 5. QUALITY OF SERVICE IN HANDOVER MANAGEMENT

---

unless when the transfer of reservation fails; in this case, option 2 is executed to enable a new path reservation.

In reactive handover, the mobile node performs reservation with the new DRM concurrently with the SIP application signaling for QoS. The DRM establishes a path from its domain to the correspondent node's domain, and performs reservation with the DRMs of the domains that the path passes through. The old reservation will be terminated after a time-out.

For particular details of the reservation protocol implementations, please refer to [16].

Integrating this network reservation and SIP application signaling, a complete QoS solution is formed to enable smooth application transition during handovers. Adaptive applications use the information from SIP application signaling to dynamically adjust their parameters to suit the new QoS capabilities. The network reservation guarantees that the QoS is maintained end-to-end from the mobile node to the correspondent node.

Figure 5.3: Illustration of the network infrastructure considered in the mobility scenario.

5.2 QoS Framework Implementation

VHM is implemented in Linux kernel 2.6.15. The Linux distribution is Ubuntu 6.10 (Edgy Eft) [23]. We build our QoS framework based on the existing TMSP, which is implemented in Python. In this project, we implement the *end-to-end Application SIP QoS signaling*. The *end-to-end network QoS signaling* is left for future work when the reservation architecture implementations proposed in [16] and [14] are available.

TMSP agents running in the mobility hosts, the correspondent node and the mobile node, and the SIP server are both extended to support the end-to-end Application SIP QoS signaling. When mobility hosts register with a SIP server, they also inform the server about its current QoS setting. Hence, the SIP server acts not only as a address directory but also as a QoS directory. We extend the TMSP's SIP REGISTER message with a QoS field so that the mobile node can just send one message for registration of

Table 5.1: SIP REGISTER message with QoS.

REGISTER	sip:192.168.1.4 SIP/2.0
Via	: SIP/2.0/UDP 192.168.2.99:15000
Max-Forwards	: 70
From	: <sip:mn@192.168.1.4>
To	: <sip:mn@192.168.1.4>
Contact	: <sip:mn@192.168.2.99:15000>;transport=udp
Expires	: 300
User-Agent	: TMSP
Call-ID	: 1722703560.76
CSeq	: 1823 REGISTER
QoS	: 1000Kbps
Content-Length	: 0

both IP and QoS (instead of two messages: SIP REGISTER and SetQoS messages). Table 5.1 represents a sample of SIP REGISTER message with QoS. The QoS setting is the host’s link access capability. The standard SIP server is still able to process the QoS-modified SIP REGISTER message since it always ignores the unknown field.

During a handoff, if the old QoS capability is a subset of (or less than) the new QoS capability, the QoS signaling is done in the *new* connection. Hence, we extend the binding-update message, which is a SIP INVITE message, with a QoS field so the mobile node can inform the correspondent node of the new IP address and new QoS in just one message. Table 5.3 shows a sample of SIP INVITE message with QoS. On the other hand, if the old QoS capability is a superset of (or greater than) the new QoS capability, the QoS signaling is done in the *old* connection. Therefore, the QoS-update message is done before the IP binding-update message². Table 5.2 represents a sample SetQoS.

Adaptive applications running at the correspondent node and the mobile node must *register* with TMSP agents for the QoS update. TMSP agent, running at the mobile node or the correspondent node, act as a QoS management agent. When the correspondent node starts communication with the mobile node, it first queries the SIP server for the

²TMSP performs the handover binding-update when the mobile node is in the new connection

CHAPTER 5. QUALITY OF SERVICE IN HANDOVER MANAGEMENT

Table 5.2: SetQoS message.

SETQOS	sip:192.168.1.4 SIP/2.0
From	: <sip:mn@192.168.1.4>
To	: <sip:cn@192.168.1.4>
Via	: SIP/2.0/UDP 192.168.2.99:15000
Contact	: <sip:mn@192.168.2.99:15000>;transport=udp
Expires	: 300
User-Agent	: TMSP
Call-ID	: 1420940831.96
Max-Forwards	: 70
QoS	: 500Kbps
Content-Length	: 0

Table 5.3: SIP INVITE message with QoS.

INVITE	sip:192.168.1.4 SIP/2.0
From	: <sip:mn@192.168.1.4>
To	: <sip:cn@192.168.1.4>
Via	: SIP/2.0/UDP 192.168.2.99:15000
Contact	: <sip:mn@192.168.2.99:15000>;transport=udp
User-Agent	: TMSP
Max-Forwards	: 70
Expires	: 300
Call-ID	: 1149353450.24
CSeq	: 154 INVITE
QoS	: 500Kbps
Content-Length	: 0

CHAPTER 5. QUALITY OF SERVICE IN HANDOVER MANAGEMENT

current mobile node's IP address and QoS capabilities using the mobile node's SIP URI. TMSP agent at the correspondent node then informs the adaptive applications. Later, when the mobile node handoffs to another network of different access capability, it would signal the correspondent node's agents of the connection change. The TMSP agent at the correspondent node again has to inform the adaptive applications of the necessary adjustment. The TMSP agent and the adaptive agents communicate with each other by Linux Socket. Figure 5.4 depicts the QoS information exchange between TMSP agent and adaptive applications. We implement a simulated video conference applications to be able to adapt to the specified QoS setting. Other applications such as: video streaming, video conferencing, can use adaptive coding to match the change in QoS settings.

Figure 5.4: Illustration of QoS Information exchange between TMSP agent and adaptive applications.

In the implementation, we assume that the mobile nodes have the capability to know the current QoS capability of its connection. The mobile nodes can perform probing or analysis on their connections to establish the QoS, as discussed in [39, 40].

5.3 Experimental Result and Discussion

In the section, we study the performance of TMSP-VHM with QoS framework and without QoS framework. Similar testbed layout as in Figure 3.11 is used. We did the experiments with 2 wireless technologies: 3G/WCDMA and 802.11 WLAN. Network A is a 802.11b WLAN; and Network B is M1's 3G network. The network components and setup are summarized in Table 3.7, Section 3.3. A simulated videoconference application at the correspondent node continuously sends traffic to the mobile node. When the mobile node switches to a new network, it will probe the access point for the current access capability. The correspondent node's adaptive applications will be informed of the access QoS change so that it can make the necessary adjustments.

5.3.1 Throughput Adaptation

This section shows the throughput sent by the correspondent node and the received throughput at the mobile node. The correspondent node transmits multimedia traffic to the mobile node at the videoconference rate of 768 Kbps to support full motion video-audio; while the mobile node handoffs across WiFi and 3G networks. Note that this rate is lower than the WLAN 802.11b bandwidth of 11 Mbps and higher than the 3G's 500 Kbps. We measure the received throughput and the packet interarrival jitter at the mobile node. Figure 5.5 and Figure 5.6 plot the case of TMSP-VHM *without* QoS framework; whereas Figure 5.8 and Figure 5.7 plot the case of TMSP-VHM *with* QoS framework. Without QoS framework, the applications at the mobile node and the correspondent node, do not inform each other of their QoS access changes; hence, they perform poorly when the mobile node moves from high-speed access WiFi network to low-speed 3G network. The correspondent node keeps sending the traffic at a high rate while the mobile node is only able to receive 500 Kbps. Furthermore, the mobile node's received throughput line is jerky with a lot of spikes. When the mobile node handoffs back to the WiFi network, there is a jump in the throughput. It is because those buffered packets at 3G network are still arriving at the mobile node after the 3G-to-WiFi handover resulting in a sudden soar in the throughput.

The throughput difference between the correspondent node and the mobile node constitutes the lost packets dropped by the 3G network. With our QoS framework, the applications at the mobile node and the correspondent node adapt better to their current connections. When the mobile node is at the high-speed 802.11b WLAN network, the correspondent node takes advantage of the connection by sending at a full-motion-video-audio rate of 768 Kbps. Whereas, when the mobile node is at the low-speed 500-Kbps 3G network, the correspondent node reduces the rate to 384 Kbps by applying a lower-compression rate. Real-time multimedia applications would perform better if they know

CHAPTER 5. QUALITY OF SERVICE IN HANDOVER MANAGEMENT

Figure 5.5: Without QoS Framework: The videoconferencing application at the correspondent node is not able to match the mobile node’s current link access capability.

Figure 5.6: Without QoS Framework: The throughput received at the mobile node is distorted with high percentage of lost traffic.

CHAPTER 5. QUALITY OF SERVICE IN HANDOVER MANAGEMENT

Figure 5.7: With QoS Framework: The adaptive videoconferencing application at the correspondent node dynamically changes media compression-rate to match the mobile node’s current link access capability.

Figure 5.8: With QoS Framework: The throughput received at the mobile node when the adaptive videoconferencing application at the correspondent node dynamically changes compressed bit rate to match the current mobile node’s link access capability.

CHAPTER 5. QUALITY OF SERVICE IN HANDOVER MANAGEMENT

---

the current access capabilities of each other. In order to enhance the user-perceived QoS, adaptive applications can enable or disable the enhanced-quality micro streams or apply different encodings.

5.3.2 Jitter Adaption

Jitter is the statistical variance of the packet interarrival time. The RFC 1889 [41] defines interarrival jitter J as the mean deviation (smoothed absolute value) of the difference D in packet spacing at the receiver compared to the sender for a pair of packets. The interarrival jitter is generally caused by slow-speed links or network congestion. Real-time applications, such as videoconference, VoIP applications, are very sensitive to the jitter and usually have user-perceived quality problems due to this effect. Thus, the jitter should be kept as minimal as possible. According to RFC 1889 [41], interarrival jitter is calculated as follows: If S_i is the sending time of packet i , and R_i is the time of arrival in RTP timestamp units for packet i , then for two packets i and j , $D(i, j)$ may be expressed as:

$$D(i, j) = (R_j - R_i) - (S_j - S_i) = (R_j - S_j) - (R_i - S_i) \quad (\text{Eq. 5.1})$$

The interarrival jitter is calculated continuously for each data packet i using this difference D for packet i and previous packet $i-1$, in order of arrival (not necessarily in sequence), according to the formula:

$$J = J + (|D(i-1, i)| - J)/16 \quad (\text{Eq. 5.2})$$

In our experiment, the correspondent node transmits to the mobile node when the mobile node handoffs from the WiFi network to the 3G network. The interarrival jitter is then calculated at the mobile node continuously after every 5000 packets according to Eq. 5.2 and Eq. 5.1. Figure 5.10 and Figure 5.9 show cases with QoS framework and without QoS framework. In both figures, the mobile node handoffs from the WiFi network to the 3G network at the packet number 25000. In Figure 5.9, when the mobile node is at the WiFi network, the correspondent node transmits at the rate of 768 Kbps; when the

CHAPTER 5. QUALITY OF SERVICE IN HANDOVER MANAGEMENT

mobile node is at the 3G network, the correspondent node adjusts the rate to 384 Kbps by applying a lower-compression encoding rate. Thus, before and after the WiFi-3G handoff, the interarrival jitter fluctuates around 7 ms with variance of less than 1 ms. In Figure 5.10, when the mobile node handoffs to the 3G network, the correspondent node is uninformed of the mobile node’s access capabilities, and continues to transmit at the same rate of 768 Kbps. The jitter line jumps to and remains at 11 ms after the 3G-WiFi handoff. Multimedia applications which are highly intolerant of the interarrival jitter will usually take steps to remove or at least reduce the jitter by buffering the arriving packets; however this is done at the expense of adding additional fixed delay. Although, it may be argued that even without the QoS framework, the application at the correspondent node may later be able to detect the mobile node’s QoS capability from the mobile node’s *receive report* and does the necessary steps to reduce the jitter. Nevertheless, the QoS framework speeds up the process to support smooth and seamless handoffs.

Figure 5.9: With QoS Framework: The Interarrival Jitter when the correspondent node transmits at the rate of 768 Kbps when the mobile node is at the WiFi network and at the rate of 384 Kbps when the mobile node is at the 3G network.

CHAPTER 5. QUALITY OF SERVICE IN HANDOVER MANAGEMENT

Figure 5.10: Without QoS Framework: The Interarrival Jitter when the correspondent node transmits at the rate of 768 Kbps when the mobile node is at the WiFi network and the 3G network.

Figure 5.11: Illustration of effects of compression rate on the interarrival jitter.

CHAPTER 5. QUALITY OF SERVICE IN HANDOVER MANAGEMENT

With the QoS framework, the videoconference applications at the mobile node and the correspondent node are better adapted to their current connections. When the mobile node is at the 11Mbps 802.11b WLAN network, the correspondent node takes advantage of this high-speed, low-delay connection by sending to the mobile node at a high rate of 768 Kbps. This rate is equal to the full motion video and audio rate. Whereas, when the mobile node is at the low-speed, high-delay 500Kbps 3G network, the correspondent node's videoconference application adjusts the sending rate by applying a lower-compression rate of 384 Kbps. Figure 5.11 illustrates the effects of compression rates on the interarrival jitter. Lower-compression multimedia packets take a shorter transmission time and have wider inter-packet intervals. This relaxes the stringent demand on the network in terms of packet delay variation requirement and increases the chance that the network is able to meet the application QoS requirements.

5.4 Summary

In this chapter, we develop a QoS framework in TMSP-VHM to enable adaptive applications to dynamically change their attributes according to the capability of the current connection. Dynamic adaptation is necessary to maintain users' perceived QoS due to the variability of wireless access technologies. Our QoS framework provides: (1) a signaling channel for applications to inform each other of the access capability changes, (2) a signaling channel for applications to reserve network resource in advance so that as QoS is assured not only *before* but also *after* network handoffs. Our QoS framework reduces the signaling cost to a minimum because it combines the QoS signaling with handover signaling. We implement the QoS framework in Python programming language. The QoS framework is tested in the real network environment and is proven to work efficiently.

Chapter 6

Conclusions and Future Work

6.1 Conclusion

Fourth generation wireless networks are expected to be overlapped networks of heterogeneous technologies such as: 3G, WiFi, WiMAX, Bluetooth. In such networks, a mobile device with multiple access interfaces should be able to roam freely between networks of different types. Many mobility management schemes have been proposed in the literature such as TMSP, MIP. However, these schemes are not sufficient in supporting smooth and fast transition across networks, which is essential for the future networks with many multimedia and real-time applications. In this project, we proposed a link-layer handover manager: Virtual Handover Management to provide multiple-interface mobile devices with fast and seamless handoffs across both heterogeneous and homogeneous networks. VHM is not a standalone mobility management scheme; rather it is an extension to the existing mobility-support protocols such as TMSP and MIP. We choose to integrate VHM into TMSP, resulting in TMSP-VHM, because TMSP enjoys advantages of simpler deployment and less overheads compared to MIP. We have run a number of experiments to test the performance of TMSP-VHM. TMSP-VHM is tested with two testbeds: *manual* Bluetooth-WiFi testbed and *real* 3G-WiFi testbed. The results indicate that TMSP-VHM provides excellent network transition experience for

CHAPTER 6. CONCLUSIONS AND FUTURE WORK

---

mobile devices. The connection is seamless and uninterrupted during the handoffs. The handover delay is minimal and close to the packet interarrival time.

In this thesis, we have also introduced a QoS framework to maintain users' perceived QoS during network transition. Our QoS framework is developed and tailored for TMSP-VHM where minimum network infrastructure support is required. A QoS framework is necessary for QoS provisioning during network handovers. With QoS signaling, adaptive applications can better use the new connection's QoS capabilities. Using our QoS framework, QoS-aware applications can seamlessly update their QoS parameters to suit the new connection's QoS capabilities during a handover. Considering that the wireless access technologies greatly vary in throughput, delay and coverage, mobility support and mobility QoS framework are important to maintain seamless, smooth connections and also to maintain users' perceived QoS. As the needs of users vary, we develop modular mobility scheme and QoS framework to cater for the individual needs. The *application signaling* and the *network signaling* are made to be independent, thus users can still take advantages of the QoS framework without having to perform network reservation. We have provided descriptions and explanations for both proactive and reactive handovers, which require different methods to support. Proactive handovers provide greater flexibility and QoS as they have no broken connection period; however, they require two interfaces to perform. Reactive handovers have a period of lost connection, which depending on the scenario, can be minimized. The QoS framework is implemented in Python [26] and tested in the 3G-WiFi testbed. The experiment results show that our QoS framework helps QoS-aware applications perform better in terms of throughput and jitter adaptation.

6.2 Future Work

Having multiple access interfaces, either in fixed or mobile nodes can bring various benefits, not just for the fast handoff purpose. These could be: (1) load sharing and load balancing, (2) redundancy and fault tolerance, (3) bi-casting, (4) permanent and (5) uninterrupted access. In this thesis, we proposed a link-layer handover management scheme, Virtual Handover Management, to manage the handoff process when mobile devices roam across heterogeneous networks. Our scheme takes advantage of the multiple-interface property to provide mobile devices a permanent and uninterrupted access, which is the fourth benefit. The other three benefits of multiple interfaces are not addressed. We propose future research in **load balancing with multiple access interfaces**. Our handover management scheme considers the case where only one access interface is active at a time. The other interface is activated only when a network handoff is needed. This wastes the unused network interface and does not make full use of the available networks. With concurrent accesses to the Internet, a mobile node can speed up its connection. However, there is an implementation issue in the current network stack since UDP/TCP sessions can be associated with only one network interface at one time. We suggest a solution to this problem by using our Virtual Device Driver. The VDD could be modified to associate with two network interfaces at a time. The upper network layers see the VDD as the sole interface; while VDD makes use of the real interface for transmission and reception. VDD balances the traffic with reference to a built-in policy. For instance, the policy would favor more traffic to and from high-speed WiFi interface rather than low-speed UMTS interface.

In this thesis, we introduce a solution to QoS provisioning in heterogeneous networks with our QoS framework. We have implemented the QoS framework with *end-to-end SIP application signaling*, while the **end-to-end network QoS signaling** is left for the future work. The end-to-end network QoS signaling implementation requires the

CHAPTER 6. CONCLUSIONS AND FUTURE WORK

---

reservation schemes' codes [14], [15]. We propose that by integrating our QoS signaling framework with the reservation architectures. A complete QoS provisioning solution is formed to enable smooth session transitions across heterogeneous networks.

6.3 Summary

In this chapter, we conclude the thesis with two main contributions. The thesis introduces a link-layer handover management scheme, Virtual Handover Management for mobile devices to roam uninterruptedly and seamlessly across networks of heterogeneous technologies by supporting multiple access interfaces. Virtual Handover Management supports both IPv4 and IPv6. The thesis also introduces a QoS framework to provide QoS provisioning in the current mobility scenario. We also discuss some proposals for the future work. The first proposal is to extend our Virtual Device Driver in supporting load balancing with multiple access interfaces. The second proposal is to integrate our QoS framework with the mobile network reservation scheme to provide complete solution to the QoS problem in mobility scenario.

Author's Publications

- [17] J.W. Tantra, N.S. Mai, D.N. Dang, T.M. Lim, C.K. Yeo, B.S. Lee, *Seamless Mobility Across Heterogeneous Wireless Networks*. In IEEE Consumer Communications and Networking Conference, USA, 2008.
- [38] J.W. Tantra, N.S. Mai, T.M. Lim, B.S. Lee, C.K. Yeo, *QoS in Terminal-assisted Mobility*. In International Conference on Information, Communications and Signal Processing, Singapore, 2007.
- [42] N.S. Mai, C.K. Yeo, B.S. Lee, D.N. Dang, *Link-layer mobility scheme for seamless handoffs across heterogeneous networks*, submitted to Journal of Network and Computer Applications, 2008.
- [18] N.S. Mai, D.N. Dang, Y. Xia, C.K. Yeo, B.S. Lee, *Seamless Handover Across Heterogeneous Wireless Networks*. In NTU InfoComm Cluster Seminar, Singapore, 2008.

References

- [1] *IEEE 802.11*. <http://www.ieee802.org/11/>.
- [2] *IEEE 802.11abg*. http://en.wikipedia.org/wiki/IEEE_802.11.
- [3] *Wireless@SG*. <http://www.ida.gov.sg>.
- [4] *WiMAX*. <http://www.wimaxforum.org>.
- [5] *Bluetooth*. <http://www.ieee802.org/15/>.
- [6] Internet Engineering Task Force. *IP Mobility Support for IPv4*, 2002.
<http://www.ietf.org/rfc/rfc3220.txt>.
- [7] B.S. Lee, T.M. Lim, C.K. Yeo, and Q.V. Le. *A Mobility Scheme for Personal and Terminal Mobility*. In IEEE International Conference on Communications, Scotland, 2008.
- [8] Internet Engineering Task Force. *SIP: Session Initiation Protocol*, 2002.
<http://www.ietf.org/rfc/rfc3261.txt>.
- [9] IEEE 802.21. *Media Independent Handover*, March 2004.
<http://www.ieee802.org/21/>.
- [10] Internet Engineering Task Force. *IP Mobility Support in IPv6*, June 2004.
<http://www.ietf.org/rfc/rfc3775.txt>.
- [11] Network Working Group. *Enhanced Route Optimization for Mobile IPv6*, May 2007.
<http://www.ietf.org/rfc/rfc4866.txt>.

REFERENCES

- [12] T.M. Lim, J.W. Tantra, B.S. Lee, , C.K. Yeo, and Y. Xia. *Terminal-assisted Network Mobility Management*. In IEEE Wireless Communications and Networking Conference, Hong Kong, 2007.
- [13] Network Working Group. *Resource ReSerVation Protocol (RSVP)*, 1997. <http://tools.ietf.org/html/rfc2205>.
- [14] A. Talukdar, B. Badrinath, , and A. Acharya. *MRSVP: a resource reservation protocol for an integrated services network with mobile hosts*. Wireless Networks, 7(1):5–19, January 2001.
- [15] Bongkyo Moon and H. Aghvami. *Reliable RSVP path reservation for multimedia communications under an IP micromobility scenario*. Wireless Communications, IEEE., 9(5):93–99, October 2002.
- [16] J. Hillebrand, C. Prehofer, R. Bless, and M. Zitterbart. *Quality-of-Service signaling for next-generation IP-based mobile networks*. IEEE Commun. Mag., 42(6):72–79, June 2004.
- [17] J.W. Tantra, N.S. Mai, D.N. Dang, T.M. Lim, C.K. Yeo, and B.S. Lee. *Seamless Mobility Across Heterogeneous Wireless Domains*. In IEEE Consumer Communications and Networking Conference, USA, 2008.
- [18] N.S. Mai, D.N. Dang, Y. Xia, C.K. Yeo, and B.S. Lee. *Seamless Handover Across Heterogeneous Wireless Networks*. In NTU InfoComm Cluster Seminar, 2008.
- [19] Huiling Jia, Peng Cheng, Zhaoyang Zhang, and Shiju Li. *An Improved Adaptive Decision Scheme for Vertical Handoff in Heterogeneous Wireless Networks*. IEEE Communications, Circuits and Systems Proceedings, 2006 International Conference, 3:1816–1820, June 2006.
- [20] Wang, H.J., Katz, R.H., Giese, and J. *Policy-enabled handoffs across heterogeneous wireless networks*. IEEE Communications, Circuits and Systems Proceedings, 2006 International Conference, pages 51–60, 25-26 Feb 1999.
- [21] *VLC media player*. <http://www.videolan.org/>.

REFERENCES

- [22] *Ekiga, formerly known as Gnomemeeting*. <http://ekiga.org/>.
- [23] *Ubuntu*. <http://www.ubuntu.com/>.
- [24] Network Working Group. *The Point-to-Point Protocol (PPP)*, July 1994. <http://www.faqs.org/rfcs/rfc1661.html>.
- [25] Network Working Group. *Dynamic Host Configuration Protocol*, March 1997. <http://www.faqs.org/rfcs/rfc2131.html>.
- [26] *Python Programming Language*. <http://www.python.org/>.
- [27] *Twisted network framework*. <http://twistedmatrix.com/trac/>.
- [28] Network Working Group. *IEEE 802.11k*. <http://grouper.ieee.org/groups/802/11/Reports>.
- [29] *Wireless Extensions*. <http://www.hpl.hp.com/>.
- [30] *WvDial*. <http://open.nit.ca/wvdial/>.
- [31] *OpenWrt*. <http://openwrt.org/>.
- [32] *MadWifi*. <http://www.madwifi.org/>.
- [33] *Bluez*. <http://www.bluez.org/>.
- [34] Network Working Group. *Internet Protocol, Version 6 (IPv6) Specification*, December 1998. <http://www.faqs.org/rfcs/rfc2460.html>.
- [35] Network Working Group. *IPv6 Stateless Address Autoconfiguration*, December 1998. <http://www.ietf.org/rfc/rfc2462.txt>.
- [36] Network Working Group. *Neighbor Discovery for IP Version 6 (IPv6)*, December 1998. <http://www.faqs.org/rfcs/rfc2461.html>.
- [37] Network Working Group. *An Ethernet Address Resolution Protocol*, November 1982. <http://www.ietf.org/rfc/rfc826.txt>.

REFERENCES

- [38] J.W. Tantra, N.S. Mai, T.M. Lim, B.S. Lee, and C.K. Yeo. *QoS in Terminal-assisted Mobility*. In International Conference on Information, Communications and Signal Processing, Singapore, 2008.
- [39] M. Li, H. Zhu, I. Chlamtac, , and B. Prabhakaran. *End-to-end QoS framework for heterogeneous wired-cum-wireless networks*. Wireless Network, 12(4):439–450, July 2006.
- [40] G. Bianchi. *Performance analysis of the IEEE 802.11 distributed coordination function*. IEEE J. Select. Areas Commun., 18(3):535–547, March 2000.
- [41] Network Working Group. *RTP: A Transport Protocol for Real-Time Applications*, January 1996. <http://www.ietf.org/rfc/rfc1889.txt>.
- [42] N.S. Mai, C.K. Yeo, B.S. Lee, and D.N. Dang. *Link-layer mobility scheme for seamless handoffs across heterogeneous networks*. submitted to Journal of Network and Computer Applications, 2008.