Mobility support and bandwidth aggregation in heterogeneous wireless networks

Zhong, Feng

2012

Zhong, F. (2012). Mobility support and bandwidth aggregation in heterogeneous wireless networks. Doctoral thesis, Nanyang Technological University, Singapore.

<https://hdl.handle.net/10356/48907>

<https://doi.org/10.32657/10356/48907>

Mobility Support and Bandwidth Aggregation in Heterogeneous Wireless Networks

ZHONG FENG

**School of Computer Engineering
Nanyang Technological University
Singapore**

A thesis submitted to Nanyang Technological University
in fulfillment of the requirement for the degree of
Doctor of Philosophy

August, 2011

*To Dad and Mom,
for their encouragements and love.
To Professor Lee Bu Sung Francis and Professor Yeo Chai Kiat
for their guidance and kind help.*

Acknowledgements

First of all, I would like to express my heartfelt thanks to my supervisors Professor Lee Bu Sung, Francis and Professor Yeo Chai Kiat. They have provided me beneficial guidance and valuable comments on my research work. Besides, they always helped me clear the doubts and encourage me to rise up to the many challenges in my research, which have helped train me into a matured researcher.

Secondly, I want to take this opportunity to thank my colleagues for their valuable and timely support throughout my PhD candidature. The discussions among us have been very fruitful and I have learnt a lot from them.

Thirdly, I extend my utmost gratitude to my wife and my parents. Their unfailing care and support have constantly motivated me to pursue my research studies. They are always at my side when I encountered difficulties. They have given me a lot of encouragement and moral support throughout my PhD candidature.

Finally, I express my respect and admiration to all who have contributed to the wonderful experience I have had.

Contents

Acknowledgements	i
Abstract	vi
List of Figures	viii
List of Tables	xi
List of Abbreviations	xii
1 Introduction	1
1.1 Background and Challenges	2
1.1.1 Wireless Technology	3
1.1.2 Future Wireless Network	6
1.2 Motivation	10
1.3 Objectives and Scopes	13
1.3.1 Mobility Management	13
1.3.2 Bandwidth Aggregation	15
1.4 Research Approach	17
1.5 Major Contributions	18
1.6 Dissertation Organization	21
2 Literature Survey	23
2.1 IP Mobility Scheme	23
2.1.1 Global Mobility	24
2.1.2 Local Mobility	28

2.1.3	Comparison of Different Local Mobility Protocols	36
2.2	Inter-Domain Mobility Protocol	38
2.2.1	Neumann's Proposal	39
2.3	Multi-Path Scheduling	41
2.3.1	End-to-End Multipath Scheduling	42
2.3.2	Network Proxy Multipath Scheduling	45
Part I: Intra-Domain Handover		50
3	EFLoM: An Efficient Framework for Local Mobility	51
3.1	Introduction	52
3.2	Motivation and Related Work	54
3.2.1	Related Work	54
3.2.2	Motivations and Goals	54
3.3	Efficient Framework for Local Mobility	55
3.3.1	System Architecture and Network Entities	55
3.3.2	Registration Procedure	57
3.3.3	Packet Forwarding	58
3.4	Qualitative Comparative Evaluation	65
3.5	Performance Analysis and Simulation	67
3.5.1	Route Hops	70
3.5.2	Protocol Signaling Cost	72
3.5.3	Handover	75
3.5.4	Traffic Overhead	77
3.6	Conclusion	79
Part II: Inter-Domain Handover		80
4	Traffic Distributor Inter-Domain Mobility (TDIDM)	81
4.1	Introduction	82
4.2	Related Work	84
4.3	Traffic Distributor Inter-Domain Mobility	85
4.3.1	Communicating with MN's PFLMA	89

4.3.2	Qualified-TD-set Selection Algorithm	90
4.4	Performance Analysis and Simulation	93
4.4.1	Binding Cache Entry (BCE) Number	94
4.4.2	Transmission Delay	96
4.4.3	Handover Delay	100
4.4.4	Protocol Signaling Cost	104
4.5	Conclusion	106
Part III: Multipath Scheduling in Wireless Network		107
5	Adaptive Load Balancing Algorithm for Multiple Homing Mobile Nodes	108
5.1	Introduction	109
5.2	Related Work	111
5.3	Adaptive Load Balancing Algorithm (ALBAM)	116
5.3.1	System Model and Overview	116
5.3.2	Delay Measurement	118
5.3.3	Virtual Flowlet Division	119
5.3.4	Packet Number Estimation Algorithm	122
5.3.5	Path Selection Algorithm	133
5.4	Simulation Results	134
5.4.1	Max Flow	136
5.4.2	Out-of-Order Ratio	136
5.4.3	Splitting Errors	139
5.4.4	Transmission Time	140
5.5	Performance for UDP Flow	141
5.5.1	Max UDP Flow	143
5.5.2	Out-of-Order Packets for UDP Flow for ALBAM and OMS	145
5.6	Conclusion	146
Part IV: Conclusion and Future Work		148
6	Conclusion and Future Work	149
6.1	Conclusion	149

6.2	Future Work	151
6.2.1	QoS-Enabled Mobility	151
6.2.2	Integration of Routing, Resource Allocation and Scheduling in Future Wireless Networks	152
Author's Publications		153
Bibliography		154

Abstract

The future wireless communication system is envisioned to be an IP-based, ubiquitous system with heterogeneous wireless networks integrated, where multiple wireless technologies converge to a common infrastructure. The unified system can bring users many benefits such as the expansion of coverage and the aggregation of multiple-path bandwidth. However, before enjoying these benefits, a lot of issues need to be solved. This thesis focuses on proposing solutions for two issues: mobility management and bandwidth aggregation.

According to the service ranges, mobility management can be classified into two categories: intra-domain mobility and inter-domain mobility. Intra-domain mobility handles user's reachability within a single domain while inter-domain mobility handles outside domain. Efficient Framework for Local Mobility Management (EFLoM) is a inter-domain mobility which employs a IP swapping technology to deliver traffic. Specifically, EFLoM employs three network entities: Mobile Node (MN), Local Anchor Router (LAR) and Wireless Access Gateway (WAG). MN is the mobile device held by users. LAR is the gateway and it holds the location information of MN. WAG is an access router of the network in the domain which regulates the intra-domain connection. When a data packet arrives, LAR updates the destination IP header with MN's current location and sends to MN. On reception, MN restores the destination address and pass on to applications. Because IP swapping can remove the tunnel header, EFLoM outperform other mobility protocols in terms of route hops, protocol signaling cost, handover delay and traffic overhead.

Except EFLoM, this thesis proposes a inter-domain mobility framework named Traffic Distributor Inter-Domain Mobility (TDIDM). Through placing a new network entity named Traffic Distributor (TD) in the aggregate network, TDIDM interconnects neighboring local

domains. When MN moves out of one domain into another, TD is notified. TD then tunnels the traffic to the new domain. After the traffic arrives at the new domain, the local mobility protocol (e.g., EFLoM) takes over the responsibility to deliver the traffic. Moreover, TDIDM develops an algorithm to select a TD set. The TD set covers MN's traffic in all paths while reducing the redirection time. Experiments have been conducted to compare it with Neumann's proposal which is another proposal to handle inter-domain issues. Results show that our TDIDM is a feasible alternative for inter-domain handover, and TDIDM outperforms Neumann's proposal in terms of binding cache entry number, transmission delay and handover delay.

The third part of the thesis is a multipath scheduling algorithm, named Adaptive Load Balancing Algorithm (ALBAM). Targeting at solving adverse impacts on TCP flows, ALBAM employs four techniques. Firstly, ALBAM takes advantage of the bursty nature of TCP flows and performs scheduling at the flowlet granularity where the packet interval is large enough to compensate for the different path delays. Secondly, ALBAM develops a packet number estimation algorithm (PNEA) to predict the buffer usage in each path. With PNEA, ALBAM can prevent buffer overflow and schedule the TCP flow to a less congested path before it suffers packet loss. ALBAM can greatly reduce if not prevent buffer overflow and schedule the TCP flow to a less congested path before it suffers packet loss. Thirdly, ALBAM employs virtual flowlet division (VFD) to divide the flow into several virtual flowlets. The VFD enables ALBAM to improve the scheduling performance when the flowlets are missing. Lastly, to obtain more accurate path delay measurement, ALBAM makes use of the Access Points (AP) to provide statistical delay measurements in each path. Simulation results show that ALBAM can yield a more accurate delay reading. Through these techniques, ALBAM can provide better performance for TCP connections than its counterparts, namely, Flowlet Aware Routing Engine (FLARE) and Opportunistic Multipath Scheduling (OMS).

List of Figures

1.1	The Deployment Model of Future Wireless Network	7
1.2	The Interworking Architecture of Future Wireless Network	8
1.3	The Protocol Stack of Future Wireless Network	10
1.4	Local Mobility vs. Global Mobility	16
2.1	Mobility IP Protocol	25
2.2	Host Identity Protocol	26
2.3	Terminal Mobility Support Protocol	27
2.4	The Topology of Local Domain	29
2.5	Cellular IP	31
2.6	HAWAII	33
2.7	Hierarchical Mobile IPv6	34
2.8	Proxy Mobile IPv6	36
2.9	The Architecture of Neumann's Proposal	40
2.10	End-to-End Multipath Scheduling	43
2.11	Network Proxy Multipath Scheduling	46
2.12	Opportunistic Multipath Scheduling	47
2.13	Flowlet Aware Routing Engine	48
3.1	System Architecture of EFLoM	56
3.2	The Handover Procedure of EFLoM	59
3.3	Inter-Domain Traffic Illustration	60
3.4	Intra-Domain Traffic Illustration	61
3.5	WAG Acquires RCoA and LCoA of Intra-Domain Communication Peers	65
3.6	Experiment Scenario	68

3.7	Route Hops for Inter- and Intra- Domain Traffic	72
3.8	Average Route Hops	73
3.9	Protocol Signaling Cost	75
3.10	Handover Delay For UDP Flow	78
4.1	Signaling Flow For Neumann's Proposal	85
4.2	The Architecture of TDIDM	87
4.3	Signaling Flow for TDIDM	88
4.4	Experimental Scenario	95
4.5	LMA1's BCE Number	97
4.6	Illustration of Traffic Flow	99
4.7	Transmission Delay	100
4.8	Illustration of the Handover	102
4.9	Handover Delay for UDP flow	103
4.10	Handover Delay for TCP flow	104
5.1	The Condition to Avoid Out-of-Order Packets	112
5.2	The Packet Interval vs Burst Interval	113
5.3	The Flowlet Interval During The Lifetime of TCP Flow	115
5.4	The System Model of ALBAM	117
5.5	Delay Measurement	120
5.6	Illustration on Virtual Flowlet Division	121
5.7	Packet Number Estimation Algorithm	123
5.8	Buffer Overflow Prediction	129
5.9	Prediction Correctness	130
5.10	Prediction Efficiency	131
5.11	Prediction Accuracy	132
5.12	Buffer Size for PNEA	133
5.13	Experiment Scenario	137
5.14	Single Path Capacity for TCP Flow	138
5.15	Maximum Flow	139
5.16	Out-of-Order Ratio When TCP is in Unstable State	140

5.17 Out-of-Order Ratio When TCP is in Stable State	141
5.18 Splitting Error When TCP is in Unstable State	142
5.19 Splitting Error When TCP is in Stable state	143
5.20 Transmission Time	144
5.21 Single Path Capacity for UDP Flow	145
5.22 Max UDP Flow and Out-of-Order Ratio vs VFD interval	146
5.23 Out-of-Order Ratio for UDP Flow	147

List of Tables

1.1	Specification of Wireless Technologies	4
2.1	Comparison of Different Local Mobility Protocols	37
3.1	Qualitative Comparison of HMIPv6, PMIPv6 and EFLoM	67
3.2	Protocol Packet Size	74
4.1	An Example of Advertisement Table	91
4.2	An Example of Path Table	91
4.3	An Example of Latency Table	92
4.4	Protocol Signaling Cost	106

List of Abbreviations

ABBREVIATIONS	FULL EXPRESSIONS
AAA	Authentication, Authorization and Accounting
ACK	Acknowledgement
ALBAM	Adaptive Load Balancing Algorithm
AN	Access Network
AP	Access Point
AR	Access Router
ASN/GW	Access Service Network/Gateway
BCE	Binding Cache Entry
BS	Base Station
BTS	Base Transceiver Station
BU	Binding Update
CIP	Cellular IP
CN	Corresponding Node
CoA	Care-of-Address
DGR	Domain Gateway Router
EFLoM	Efficient Framework for Local Mobility Management
ER	Edge Router
ETSI	European Telecommunications Standards Institute
FA	Foreign Agent
FLARE	Flowlet Aware Routing Engine
FWN	Future Wireless Network
GFA	Gateway Foreign Agent
GGSN	Gateway GPRS Support Node
GPRS	General Packet Radio Service
GSM	Global System for Mobile Communications
HA	Home Agent
HARQ	Hybrid Automatic Repeat-Request
HIP	Host Identity Protocol

HIT	Host Identity Tag
HLMA	Home Local Mobility Anchor
HLR	Home Location Register
HMIPv6	Hierarchical Mobile IPv6
HoA	Home Address
HSDPA	High-Speed Downlink Packet Access
LAR	Local Anchor Router
LBA	Local Binding Acknowledgement
LBU	Local Binding Update
LCoA	Own-link Care-of-Address
LMA	Local Mobility Anchor
LTE	Long Term Evolution
MAG	Mobile Access Gateway
MAP	Mobility Anchor Point
MIF	Multiple Interface
MIMO	Multiple-Input Multiple-Output
MIP	Mobile IP
MN	Mobile Node
MR	Mobile Router
NEMO	Network Mobility
NFLMA	New Foreign Local Mobility Anchor
OMS	Opportunistic Multipath Scheduling
PBA	Proxy Binding Acknowledgement
PBU	Proxy Binding Update
PDA	Personal Digital Assistant
PDN	Packet Data Network
PFLMA	Previous Foreign Local Mobility Anchor
PMIPv6	Proxy Mobile IPv6
PNEA	Packet Number Estimation Algorithm
PTD	Pivot TD
RA	Router Advertisement
RBU	Regional Binding Update
RCoA	Regional Care-of-Address
RNC	Radio Network Controller
RO	Route Optimization
RR	Register Request
RS	Router Solicitation
RtrAdv	Router Advertisement

SE	Splitting Error
SGSN	Serving GPRS Support Node
SIP	Session Initiation Protocol
SIP-NEMO	Session Initiation Protocol Network Mobility
TD	Traffic Distributor
TDIDM	Traffic Distributor Inter-Domain Mobility
TMSP	Terminal Support Mobility Protocol
TNEMO	Terminal-Assisted network mobility
VFD	Virtual Flowlet Division
VoIP	Voice over IP
WAG	Wireless Access Gateway
WLAN	Wireless Local Area Network
WMAN	Wireless Metropolitan Area Network
WPAN	Wireless Personal Area Network
WRR-JA	Weighted Round Robin-Jump Ahead
WWAN	Wireless Wide Area Network

Chapter 1

Introduction

Wireless industry has witnessed a rapid growth in the past decades. Since the launch of the first automated cellular network (1G network) in 1979, the number of wireless subscribers has increased thousands of times. According to one market survey conducted by Ericsson, by the end of 2015, the subscriber number will increase to 7-8 billion [1].

The drive behind the rapid growth of wireless technology is its advantages over fixed line. Unlike traditional wired network, wireless network allows users to access to the Internet without the existence of physical cabling. Therefore, in wireless networks, users can move within a large range without the constraint of wired lines. This enhances the mobility support for users. Nowadays, users are getting used to call their friends or log on to the Internet with a smartphone while they are moving. In addition, cable-free infrastructure brings wireless network the advantage of easy deployment and maintenance. If users want to set up a wireless network in the office or at home, they do not need to lay the cable. After they power up the wireless router and connect the router to the backhaul Internet, the wireless network is ready for use. Without the need to lay wired lines, the cost of setting-up wireless networks is reduced.

Owing to the benefits of wireless technology, wireless networks have enjoyed wide deployment. The success in the market, in turn, drives the prosperous development of wireless technology. In the past decades, a lot of wireless technologies have been proposed and implemented. According to the origin of these wireless technologies, they can be classified into two types. One is non-cellular networks which are developed from packet data network (PDN). These non-cellular networks mainly provide high-rate data service and users usually employ them to access the Internet. For example, nowadays, WiFi is widely deployed in public places such as airports and cafes to provide access service to users. The other kind of wireless network is cellular network which is developed from circuit network. At first, the goal of cellular network is to provide person-to-person call service to users just as the telephone. Nowadays, with the increasing requirement on data service, they are evolving to provide data service. For example, the 2G network (e.g., GSM) does not support data service, but its successor 3G network (e.g., UMTS) adds data service to as high as 10Mbps into the network.

1.1 Background and Challenges

The future wireless communication system is envisioned to be an IP-based, ubiquitous system with heterogeneous wireless networks integrated [2, 3]. All services (e.g., voice service and data service) are delivered with IP packets. As the IP network has a higher utilization of bandwidth than circuit network, the throughput of future system is expected to be improved. In addition, the future system will integrate existing heterogeneous wireless networks and provide a unified access to users. With a mobile phone, users can roam anywhere even if they are covered by different kinds of wireless technologies. In the areas with overlapped multiple wireless networks coverage, users can access them simultaneously and thus aggregate their bandwidth.

1.1.1 Wireless Technology

Here, an introduction on the existing wireless technologies which forms the base of the future system is presented. There are multiple kinds of wireless technologies. Different technologies have different performances. Depending on their coverage area, these wireless technologies can be classified into four types. Wireless Personal Area Network (WPAN), Wireless Local Area Network (WLAN), Wireless Metropolitan Area Network (WMAN) and Wireless Wide Area Network (WWAN). The four kinds are listed in ascending order of their coverage. WPAN's coverage is the smallest. Its reach is around a person's body, usually a few meters. In practice, WPAN is usually used to interconnect personal devices such as telephones and Personal Digital Assistants (PDA). WLAN's coverage is larger than WPAN and it is usually deployed in small room such as homes, offices and airport. Since the transmission range of WLAN can reach 100 meters, it usually serves as an alternative to cables and provides an access to Internet. The third kind of wireless technology is WMAN which expands the coverage to a campus or a city. Compared with WPAN and WLAN, WMAN has higher data-rates and more mobility support. In a WMAN, users can keep their connection unbroken while moving from one place to another. However, the movement is limited to a city. If users want to roam farther to another city, WWAN is required. In contrast to the other three kinds of wireless technologies, WWAN can provide wireless access nationwide or even globally. With the help of WWAN, users can enjoy communication service anytime, anywhere as long as there is WWAN radio signal.

Table 1.1 lists some common wireless technologies. These wireless technologies vary in throughput, transmission range, radio technology and frequency band. For example, Bluetooth [4] is designed for exchanging data within WPAN area. It is widely deployed between the non-resident equipments. To exchange data over Bluetooth, a device needs to be assigned as the master and other device serves as the slave. As a replacement for cabling devices, Bluetooth is widely installed in personal devices such as laptops, digital cameras

Table 1.1: Specification of Wireless Technologies

Technologies	Bluetooth	WiFi	WiMAX	GPRS	HSDPA
Standards	IEEE 802.15	IEEE 802.11	IEEE 802.16	3GPP Rel97	3GPP Rel5-9
Network Type	WPAN	WLAN	WMAN	WWAN(2G)	WWAN(3G)
Frequency Band	2.4GHz	5/2.4GHz	2.5GHz	0.9/1.8GHz	2.1GHz
Throughput	1Mbps	11Mbps	20Mbps	114kbps	10Mbps
Radio Technology	FHSS	OFDM /MIMO	MIMO-SOFDMA	FDMA & T-DMA	CDMA/FDD /MIMO
Trans Range	10m	100m	50km	35km	35km

and mobile phones.

However, the transmission range of Bluetooth is usually no more than 10 meters. The short range restricts the application of Bluetooth. If the users want to gain wireless access within a wider area, the WiFi [5] technology which can cover around 100 meters provides a valuable option. Unlike Bluetooth's master-slave infrastructure, WiFi employs a Carrier Sense Multiple Access with Collision Avoidance (CSMA/CA) model to connect all the members in its network. CSMA/CA is a loose topology and it does not require the connection setup as Bluetooth. The users can engage and leave WiFi network at any time they want. Moreover, the throughput of WiFi network is higher than Bluetooth. WiFi enjoys a popular deployment all over world. Every year, there are about 800 million new WiFi devices being produced.

Generally, WiFi is designed for small rooms and it is unsuitable for connection on the move. When users take a train or bus, they will have to turn to other wireless technology such as WiMAX [6–8]. WiMAX is short for Worldwide Inter-operability for Microwave Access and it is originally designed as an alternative to wired access networks such as fiber-optic links, cable TV and DSL. In subsequent development, WiMAX has incorporated mobility support since the standard of 802.16e was established. The transmission range of WiMAX

network is measured in kilometers, which is much larger than WiFi. At the same time, WiMAX provides inherent Quality-of-Service (QoS) support in the transmission which enables WiMAX to be applied in many scenarios such as voice-over-IP (VoIP) and video connections.

Bluetooth, WiFi and WiMAX are all developed from packet-switched networks and they are more suitable for data communications. In current deployment, the family of cellular networks takes a big portion. Unlike non-cellular networks, the development of cellular networks is divided into three generations. The first generation of cellular network (1G) was launched in around 1980 and it only provided voice service. From 2G network, cellular networks began to provide packet switch support. General Packet Radio Service (GPRS) [9] is the first cellular network to allow packet data transmission. GPRS is an extension of Global System for Mobile communications (GSM) which is a 2G network developed by European Telecommunications Standards Institute (ETSI). When ETSI designed the GSM standard, data packet capability was not taken into consideration. Hence, GSM does not support packet switch. GPRS extends GSM by using eight time slots in a 200 KHz bandwidth spectrum. The data rate of GPRS is as high as 114 Kbps.

When cellular network enters the 3G development, the data rate got a huge boost. High-Speed Downlink Packet Access (HSDPA) [10] is one example of 3G network. As the successor of GPRS network, HSDPA can achieve a data rate as high as 10 Mbps. In order to improve the data transfer capacity, HSDPA employs advanced antenna technology such as Multiple-Input Multiple-Output communications (MIMO), Hybrid automatic repeat-request (HARQ) and adaptive modulation and coding. Currently, HSDPA is widely deployed all over the world such as Singapore, Australia and China.

1.1.2 Future Wireless Network

As a successor of existing wireless technologies, the Future Wireless Network (FWN) is expected to be a heterogeneous system that combines the advantages of existing wireless technologies. To evolve towards the convergence of these technologies, FWN builds an IP infrastructure which has many advantages. Firstly, IP is a loose framework and it can easily glue multiple wireless technologies together [11, 12]. Under IP infrastructure, different wireless technologies develop independently and converge at the IP layer to form a unified system. Secondly, IP technology enhances the transmission rate. A lot of research works [13–16] have proven that packet-switch can provide better bandwidth sharing. This is important because of the limited throughput resource in wireless networks. Thirdly, it is cost effective to deliver data over the common IP infrastructure. As IP technology allows the integration of different wireless technologies, it is unnecessary to build different network for each kind of technology.

Actually, the FWN can be introduced in three aspects, namely, deployment, interworking architecture and protocol stack. Figure 1.1 presents the deployment model of FWN. In the FWN, different wireless technologies have different deployment and coverage. They overlap and complement each other to form an enhanced access system. In Figure 1.1, the 3G cellular network is widely deployed in an urban environment with the hexagon form. Among the cellular coverage, the WiFi networks serve as scattered hotspots to provide higher data connections to users. In the integration of cellular network and WiFi network, users can enjoy a service range wider than WiFi and a bandwidth occasionally higher than 3G network. Moreover, the WiMAX network can provide access in isolated areas which cannot be covered by cellular network, such as rural areas and remote neighborhoods. Those wireless networks together provide a new application environment to users. In the new environment, users can switch to any kind of wireless networks and enjoy a better experience.

Figure 1.1: The Deployment Model of Future Wireless Network

Figure 1.2: The Interworking Architecture of Future Wireless Network

The second model is the interworking architecture which is presented in Figure 1.2. The FWN employs the loose-coupled model which has been proven to be a highly efficient and economical model for integrating multiple technologies. In Figure 1.2, the network components for each type of wireless technology are listed. In the WiMAX network, Base Station (BS) provides wireless signal to users and the Access Service Network of Gateway (ASN-GW) connects the BS to IP core network. The topology of WiFi network is similar to WiMAX network, both of which are developed from the packet-switch system. The second kind of network topology is cellular network such as GPRS, UMTS and CDMA2000. Take UMTS network for example, there are four kinds of components in the network. NodeB is responsible for providing wireless signal. The Radio Network Controller (RNC) is responsible for radio resource management. The Serving GPRS Support Node (SGSN) is responsible for

the delivery of data packets from and to the mobile stations within its geographical service area. SGSN connects RNC and Gateway GPRS Support Node (GGSN) serves as a router for the UMTS network. In addition, IP core network has connections with services such as Home Location Register (HLR), Authentication, Authorization and Accounting (AAA) and Billing. These services can be used by these multiple networks together.

Despite the different components of each wireless network, they can be modeled as a simple one which is similar to the WiFi network. In the simple model, there are two basic components. One is the base station (BS) which provides wireless radio signal over the air to the users, examples of which are the NodeB in UMTS network, the access point (AP) in WiFi network and the base transceiver station (BTS) in CDMA2000. The other component is access router (AR) which serves as an IP router between wireless access network and core network. When the AR receives an IP packet, it locates the user and delivers the packet to the corresponding BS. The delivery mechanism differs for different types of wireless networks. From the IP traffic point of view, AR hides the infrastructure difference between these networks and implements the delivery mechanism within the access network.

The protocol stack model (Figure 1.3) of FWN has presented the independent development. In the protocol model, each wireless technology has its own implementation of the physical and data link layers [17]. All these implementations connect to the same network layer which provides a unified interface to application layer. With the separation of network layer, FWN model is divided into two parts. On one hand, the existing transport (e.g., TCP or UDP) and application protocols can be run on the network layer without any change. On the other hand, the lower layer (physical, MAC) can be developed independently. Furthermore, in FWN model, some control planes such as mobility management and connection management can be added. These control planes can use information from multiple layers in the protocol stack.

Figure 1.3: The Protocol Stack of Future Wireless Network

1.2 Motivation

Compared to existing wireless systems, the service of FWN is expected to be improved in the following aspects. Firstly, the subscribers' access reach will be expanded. Owing to the limitation of wireless transmission, current wireless networks usually do not cover a large area. Instead, they provide service in scattered hot-spots. The scattered deployment hinders the continuous connection with wireless networks. To prevent the sessions from disruption, subscribers have to stay in the neighboring hot-spots. However, in the integrated system, subscribers can move to areas which are covered by another wireless technology. For example, when a subscriber leaves a WiFi network and enters the coverage of UMTS network, he can handover from WiFi to UMTS network and continue his sessions. From this point, the integrated system has a coverage as wide as the union of the multiple wireless

networks.

The second advantage of FWN is that subscribers can employ different wireless technologies to satisfy their requirements. Existing wireless technologies are designed to provide different services and they have different performances. For example, WiFi is designed to provide an access of high throughput comparable with wired link to users while UMTS is deployed in places where the mobile support is more valued. Since future integrated system allows subscribers to access different kinds of wireless technologies, they can choose one that best fits their requirements. If subscribers are running bandwidth-hungry applications, they can connect to the WiFi network. If subscribers pay more attention to mobility, they can turn to UMTS network. From this perspective, the FWN gathers the advantages of multiple wireless networks and provides them to subscribers in a unified way.

Thirdly, future integrated system can improve the performance in terms of bandwidth. Existing networks always suffer from limited bandwidth due to bandwidth-hungry applications and heavy usage. In the FWN which integrates multiple networks, there are many places that heterogeneous networks will overlap one another. The overlapped coverage provides an opportunity to improve the users' throughput. If users can gain access to multiple networks simultaneously, they can take advantage of the bandwidth resource of multiple networks and improve their experience by aggregating them.

Although the future wireless network poses a promising advantage over existing network, the integration of multiple wireless networks is not an easy task. Several issues need to be resolved [17, 18]. Firstly, a common infrastructure for interworking different kinds of wireless networks is needed [11, 19–23]. Then, an evolution roadmap that upgrades current network to future network [3, 24, 25] is also needed. Other issues, such as the Authentication, Authorization and Accounting (AAA) mechanism between different networks [26, 27], the security check across different networks [28–30], the multiple-interface decision function [31–33], the radio and spectrum management [34–36], the mobility protocol across multiple

networks [37–42] and the bandwidth aggregation for multiple wireless links [43–45], are all crucial for a successful integration of heterogeneous networks.

Among these issues, the mobility protocol and bandwidth aggregation in heterogeneous networks are of great importance. The mobility protocol enables users to handover across different wireless technology while keeping their sessions undisturbed. In the FWN where multiple wireless technologies complement each other, the performance of mobility protocol is the key for user session continuity and service quality. If the mobility protocol can provide a unified support for the handover across heterogeneous networks, users can enjoy a combination of their coverage and service. For example, if a user finds out he has moved out of the WiFi coverage or if he finds WiFi network is congested, he can handover the connection to another available network such as 3G or WiMAX. Then, the user can avoid the network disconnection or the QoS downgrade. Another big drive for the integration of multiple wireless networks is bandwidth aggregation. Current wireless networks often impede the development of application because of their limited throughput. Unlike wired lines, wireless network's spectrum is narrow and its signal is often affected by several causes such as radio propagation loss, shadowing and multipath fading. Therefore, the effective throughput of wireless networks is much less than wired networks, often degrading to Mbps as compared to the several Gbps in wired networks. In the FWN, overlapped heterogeneous networks provide an opportunity to enhance the wireless throughput performance. If users can access multiple networks simultaneously with a multiple-interface device (e.g., smartphone), they can aggregate the throughput of heterogeneous networks and support more bandwidth-hungry applications via wireless networks.

1.3 Objectives and Scopes

Motivated by the promising benefits of the FWN, the objectives of the research reported in this dissertation are to propose solutions to address two critical issues in the FWN, namely, mobility management and bandwidth aggregation. The proposed solutions are for the FWN infrastructure which is shown in Figures 1.1, 1.2 and 1.3.

1.3.1 Mobility Management

The purpose of mobility management is to satisfy the requirement of ubiquitous service and provide the handover support when people move across networks. As the FWN converges heterogeneous networks at the IP layer, our solution is proposed at the IP layer. The IP model can free mobility protocol from the implementation of different wireless technologies and simplify the implementation of mobility protocol. The basic logic of IP mobility lies in the separation of the roles of IP address. In the existing network model, the IP address has two conflicting roles. One is the identification and the other is the location. As the identification, IP address is usually used by the upper layers to establish connection. For example, TCP uses the IP address as well as port number to establish a connection. To enable users to maintain their connection during the movement, the mobility protocols should keep the IP address unchanged during the period. However, the unchanged IP address will result in inconsistency with respect to the user's location. IP address is used by the routing protocol to determine the location of the mobile node. If users change their location, the IP address should be updated correspondingly failing which, the router cannot deliver the packets to the users. To ensure users can receive the traffic, the IP address should be changed when they move to a new location.

To solve the conflicting requirement of identification and location, mobility protocols propose a separation of the two conflicting roles of IP address such as Mobile IP and HIP,

Mobile IP [46–48] suggests to assign two addresses to each mobile node. One is Home Address (HoA) which is used as identification and the other is Care-of-Address (CoA) which plays as the role to resolve the location. HIP is a mobility protocol that adds a new layer to provide an identification function for the connection. The IP address still plays the location role. At the same time, HIP maps the two roles in the protocol stack of the end user. Chapter 2 will give a detailed introduction to these protocols.

Through separating the conflicting roles of IP address, the IP mobility protocols such as Mobile IP and HIP can help users maintain their sessions across different networks. However, these mobility protocols are not good enough for users who require short handover delay. When users move far away from its home agent or peer, the operation of updating location address may take a considerable amount of time. Thus, the users may suffer a long handover delay. In addition, the exposure of users' location will pose security issues such as tracking and eavesdropping.

To reduce the handover delay and hide the location, some researchers take advantage of network topology and classify the handover into two types: Intra-domain handover and the inter-domain handover. The intra-domain handover is a movement across neighboring networks within the same domain while inter-domain handover is a movement across domains. Corresponding to the two kinds of handover are two kinds of mobility protocols: global mobility and local mobility. Global mobility handles the inter-domain handover and local mobility handles the intra-domain handover. When users move within a single domain, they employ local mobility to maintain their sessions continued. If users move out of a domain, they need the help of global mobility to switch their traffic to the new location.

The separation of global mobility and local mobility provides a more flexible mobility solution. Global mobility is responsible for delivering traffic to the domain that users are residing right now. After the traffic arrives at a domain, local mobility is responsible for delivering traffic to the wireless links. The juncture node between local mobility and global

mobility is the Domain Gateway Router (DGR) which is shown in Figure 1.4. In Figure 1.4, if users move across different access networks (AN) (e.g., from AN2 to AN3), an inter-domain handover occurs and global mobility redirects the traffic to DGR2. When DGR2 receives the traffic, it tunnels the traffic to AR3 to which the users are attached currently. If users keep on moving to AN4, an intra-domain handover occurs. Hence, users employ local mobility to deliver the traffic to AN4.

1.3.2 Bandwidth Aggregation

The second issue in the FWN integration is bandwidth aggregation. Current wireless applications often suffer from the limited bandwidth of wireless links. The FWN environment provides an opportunity to aggregate the bandwidth of heterogeneous wireless networks. If users can access multiple networks simultaneously, they can employ a scheduling algorithm to split traffic over multiple paths proportionally and achieve a higher throughput. In addition, multipath access can enhance the network robustness. If users find one wireless link is down or disconnected, the multiple path scheduling can help user switch traffic to another healthy link. Hence multiple paths provide a certain level of redundancy to safeguard against link failure.

Given the advantages of multipath scheduling, our work has developed a new scheduling algorithm. The new scheduling algorithm is based on the FWN model and it aims to improve TCP performance over multiple wireless links. The algorithm investigates two reasons for the bad performance of TCP flow over wireless link. One is the occurrence of out-of-order packets due to the different delays in the multiple paths. The other is the packet loss which is resulted from the limited bandwidth of the wireless networks. Targeting at solving the two issues, firstly, our algorithm takes advantage of the bursty nature of TCP flows and performs scheduling at the flowlet granularity where the packet interval is large enough to compensate

DGR: Domain Gateway Router
 AR: Access Router
 AP: Access Point
 MN: Mobile Node
 AN: Access Network
 CN: Corresponding Node

Figure 1.4: Local Mobility vs. Global Mobility

for the different path delays. Secondly, our algorithm develops a packet number estimation algorithm (PNEA) to prevent the buffer overflow in the wireless links.

1.4 Research Approach

The dissertation focuses mainly on the two issues in the future wireless networks: mobility management and bandwidth aggregation. Mobility management allows users to roam universally in the unified future wireless networks and multipath scheduling can help aggregate bandwidth of multiple paths. With solutions being proposed to address the two issues, users are expected to enjoy better service in the future wireless networks.

To develop a new mobility protocol, a literature survey on the existing mobility protocols is first carried out. The survey depicts a two-level hierarchical mobility framework and two mobility protocols, one for each level, are proposed. At the local mobility level, Efficient Framework for Local Mobility Management (EFLoM) is proposed. EFLoM is based on observation of the limitations of the existing local mobility protocols which do not optimize the routing path for the intra-domain traffic and it fulfills the function of delivering intra-domain traffic via the shortest path. Evaluation shows that EFLoM can outperform other local mobility protocols, namely, HMIPv6 and PMIPv6, in terms of route hops, protocol signaling cost, handover delay and traffic overhead.

Although EFLoM is proven to yield better performance than its peer, its service range is limited to a small area. To expand the movement and connect neighboring domains together, a Traffic Distributor Inter-Domain Mobility (TDIDM) at the inter-domain mobility level is proposed. TDIDM can reduce the handover delay when users move out of one domain into another. To evaluate the performance of TDIDM, the shortcomings of existing representative inter-domain protocols are first listed. Targeting at solving these shortcomings, TDIDM, which employs aggregate routers to redirect traffic during routing is presented. Simulations

are then conducted to compare the performance of TDIDM against its counterpart. The simulation shows that TDIDM can achieve a better performance over others in terms of binding cache entry number, transmission delay and handover delay.

The research methodology for multipath scheduling work is as follows: a comprehensive literature survey on the existing multipath scheduling algorithms is carried out. The algorithms are being classified into two broad categories: end-to-end scheduling and network proxy scheduling, based on their implementation mode. Since end-to-end scheduling needs an upgrade of the protocol stack which is costly for mobile users with resource constraint, our scheduling is focused on network proxy scheduling which does not require involvement of mobile users. The limitations of existing network proxy scheduling which can reduce out-of-order packets in the wireless networks are identified. The bursty characteristic of TCP flow is investigated and a new scheduling algorithm proposed. The new algorithm can reduce if not eradicate the disorder packets. To evaluate the performance of our scheduling, a lot of simulations are conducted against other representative network proxy scheduling. Experiments show that the proposal can achieve a higher bandwidth aggregation over others.

1.5 Major Contributions

The major contributions of the dissertation are summarized as follows:

To improve the performance of existing mobility protocols, two mobility protocols named Efficient Framework for Local Mobility Management (EFLoM) and Traffic Distributor Inter-Domain Mobility (TDIDM) are proposed, both of which inherit from the idea of mobility hierarchy.

EFLoM is developed to handle the issue of intra-domain movement. It provides users with the ability to deliver traffic without any overhead while they are roaming in the domain.

Existing intra-domain mobility protocols usually employ IP tunnel to deliver traffic. The tunnel header increases the traffic cost. To remove the tunnel header, EFLoM employs a IP header swapping technology to deliver traffic. On reception of the data package, EFLoM updates the destination IP address with the receiving node's latest address. When the data package arrives, the receiving node restores the destination IP address to keep the IP change hidden from the applications. Specifically speaking, EFLoM employs three network entities: Local Anchor Router (LAR), Wireless Access Gateway (WAG) and Mobile Node (MN). MN is the mobile device held by users. LAR is the gateway of the domain and WAG is an access router of the network in the domain. When MN moves across networks within a domain, LAR holds the latest location of MN. If LAR receives traffic aiming at MN, LAR updates the IP header with MN's new IP address which can ensure the traffic is delivered to MN's new location. Moreover, to enable peers of the same domain to communicate directly, EFLoM allows WAG to intercept the intra-domain connection and redirect traffic to MN via optimal path. Analysis and simulations show that EFLoM can outperform other mobility protocols, namely, HMIPv6 and PMIPv6 in terms of route hops, protocol signaling cost, handover delay and traffic overhead.

TDIDM is an inter-domain mobility protocol. Unlike inter-domain mobility which provides service within a domain, TDIDM glues neighboring local domains together and allows MN to handover between them. TDIDM places a new network entity named Traffic Distributor (TD) in the aggregate network. The TD interconnects the local domain with the Internet as well as with the neighboring local domains. When MN moves out of one domain into another, TD is notified. Then, TD intercepts the traffic before it arrives at MN's old domain and tunnels it to the new one. After the traffic is received by the new domain gateway, the local mobility protocol (e.g., EFLoM) will take over the responsibility to deliver traffic to the correct location of MN. To choose appropriate TD among candidates, TDIDM develops a selection algorithm. The algorithm selects several TDs from candidates and forms a set. The

TD set can cover MN's traffic in all paths while reducing the redirecting time. Experiments are conducted to compare it with Neumann's proposal which is another proposal to handle inter-domain issues. Results show that our proposal is a feasible alternative for inter-domain handover, and TDIDM outperforms Neumann's proposal in terms of binding cache entry number, transmission delay and handover delay.

Besides the mobility management, the bandwidth aggregation is also an important challenge for integrating multiple wireless networks. In FWN system where heterogeneous networks overlap with one another, the simultaneous access of multiple wireless networks opens up the possibility to combine and aggregate the different wireless links to enhance bandwidth. A multipath scheduling algorithm, named Adaptive Load Balancing Algorithm (ALBAM), to distribute TCP flows across multiple wireless links is presented. Targeting at solving adverse impacts on TCP flows, ALBAM employs four techniques. Firstly, ALBAM takes advantage of the bursty nature of TCP flows and performs scheduling at the flowlet granularity where the packet interval is large enough to compensate for the different path delays. Secondly, ALBAM develops a packet number estimation algorithm (PNEA) to predict the buffer usage in each path. With PNEA, ALBAM can prevent buffer overflow and schedule the TCP flow to a less congested path before it suffers packet loss. Thirdly, ALBAM develops virtual flowlet division (VFD) to divide the flow into several virtual flowlets. VFD enables ALBAM to improve the scheduling performance when the flowlets are missing. Lastly, to obtain more accurate path delay measurement, ALBAM employs the Access Points (AP) to provide statistical delay measurements in each path. Simulation results show that the statistical method can yield a more accurate delay reading than FLARE, especially in one-hop wireless networks. The results also show that ALBAM can provide better performance for TCP connections than its counterparts, namely, FLARE and OMS.

1.6 Dissertation Organization

The dissertation comprises six chapters. Chapter 1 describes the issues of mobility management and bandwidth aggregation. For mobility management, the existing mobility management protocols are being classified into two broad categories: global mobility and local mobility. A definition is given for the two kinds of mobility and a brief comparison between them is described. For bandwidth aggregation, the benefits and opportunities that bandwidth aggregation can contribute to future wireless system are discussed.

Chapter 2 presents the literature survey on existing IP mobility protocols as well as multipath scheduling techniques.

Chapter 3 proposes a new efficient framework for Local Mobility Management (EFLoM) to handle the local mobility management for mobile nodes. Through manipulating the data packets, EFLoM enables MN to roam within a local domain. To avoid introducing traffic overhead into the local domain, EFLoM employs an IP mapping mechanism to deliver MN's traffic. All the packets' destination address is updated with MN's newest IP address by LAR before they enter the local domain. Then, the data packets are restored by WAG with MN's previous IP address and sent to MN. In addition, to evaluate the performance of EFLoM, experiments are conducted to compare with HMIPv6 and PMIPv6, both of which are local mobility management protocols. The compared metrics include route hops, protocol signaling cost, handover delay and traffic overhead. Analysis and simulations show that EFLoM provides a good alternative to manage mobility in a local domain.

Chapter 4 proposes an alternative to Neumann's proposal to allow mobile nodes to move between different local domains. With the help of a traffic distributor (TD), the neighboring domains are combined to enable mobile nodes to continue their existing sessions when they move across multiple local domains. The proposal's effectiveness is evaluated against Neumann's proposal in terms of LMA's BCE numbers, transmission delay and handover

delay. Through experiments, Traffic Distributor Inter-Domain Mobility (TDIDM) outperforms Neumann's proposal in all three aspects and proves to be a good viable alternative to Neumann's proposal.

Chapter 5 presents a new scheduling algorithm named Adaptive Load Balancing Algorithm (ALBAM) to split traffic across multiple wireless links within the ISP infrastructure. Targeting at solving the two adverse impacts on TCP flows, ALBAM develops two techniques. Firstly, ALBAM takes advantage of the bursty nature of TCP flows and performs scheduling at the flowlet granularity where the packet interval is large enough to compensate for the different path delays. Secondly, ALBAM develops a packet number estimation algorithm (PNEA) to predict the buffer usage in each path. With PNEA, ALBAM can prevent buffer overflow and schedule the TCP flow to a less congested path before it suffers packet loss.

Chapter 6 concludes the dissertation and recommends the direction for future work.

Chapter 2

Literature Survey

2.1 IP Mobility Scheme

With the increasing number of people with mobile devices who demand ubiquitous wireless access, it becomes an important issue to provide people seamless and undisturbed mobility service. IP mobility protocol implements such a service by separating the two conflicting roles of IP address - identification and location. Through several years' of development, a lot of IP mobility protocols have been proposed. These protocols, according to their different service ranges, can be broadly classified into two categories, namely, global mobility and local mobility protocols. Global mobility is a protocol that maintains the mobile node's (MN) reachability when MN roams within a wide area, whereas local mobility protocol provides mobility management to MN within a restricted area. The two categories of mobility protocols are not independent but cooperate and complement with each other. Generally, MN needs to employ both global mobility and local mobility to maintain its session continuity. If a handover happens in a local domain (e.g., MN moves from one network to another, both of which are within the same domain), local mobility will help MN connect to the Internet continuously in the new location. However, if the handover happens between different domains,

MN will need to perform global mobility operation.

2.1.1 Global Mobility

Mobile IP (MIP)

There are a lot of global mobility protocols such as Mobile IP (MIP) [46–49], Host Identity Protocol (HIP) [50, 51] and Terminal Support Mobility Protocol (TMSP) [52]. Among them, the most common one is MIP which is presented in Figure 2.1. Under MIP, each MN has two addresses - home address and Care-of-Address (CoA). The home address is the identification of MN whereas the CoA plays the location role for MN. As soon as MN is powered on, the home address is assigned to MN by its home agent. During the lifetime of MN, the home address is kept unchanged no matter where MN moves. If MN wants to communicate with its peer, the home address is used to establish connection. The invariable home address can prevent MN from reestablishing connection. At the same time, to help MN receive the traffic, MIP assigns a CoA to track the movement of MN. The CoA is topologically consistent with MN's latest location. Every time MN moves to a new network, MN updates its CoA and sends a registration request message to its Home Agent. The registration message helps Home Agent to create a mapping between home address and CoA. Since the connection is based on MN's home address, the traffic destined to MN will be firstly routed to MN's home network in which the home agent receives the traffic on behalf of MN. Then, the home agent encapsulates the traffic with MN's CoA and sends it out. As CoA is topologically correct, the traffic will be received by MN successfully. Figure 2.1 presents the workflow of MIP.

Host Identity Protocol (HIP)

Another global mobility protocol is Host Identity Protocol (HIP) [50, 53, 54] which is presented in Figure 2.2. Unlike MIP which assigns two addresses to MN, HIP introduces a

Figure 2.1: Mobility IP Protocol

host identity layer between the transport and network layers. The newly added layer holds a host identifier Tag (HIT) to free the IP address from the role of identification. Prior to MN communicating with its peer, it firstly uses HIP protocol to establish a HIP association with its peer, the Corresponding Node (CN). HIP association includes HIT pairs as well as the IP address pairs. The HIT is bounded by the transport layer to represent the host identification, and the IP address is only used for packet forwarding. If MN sends a packet to its peer, the packet is passed down by the transport layer to the HIP layer which maintains a binding between HIT and IP address. Then, HIP layer uses the IP address to update the HIT in the packet and sends it out. The IP address ensures the packet can be delivered to the receiver correctly. On the receiver side, when the packet arrives, the HIP layer restores the HIT of the packet and sends it to the upper layer. If MN moves to a new IP address, its peer will be notified of the new address and HIP will update the binding with the new address. The

Figure 2.2: Host Identity Protocol

updated binding thus help communication peers obtain mobility support on their movement.

Terminal Mobility Support Protocol (TMSP)

Terminal Mobility Support Protocol (TMSP) is one example protocol that employs Session Initiation Protocol (SIP) [55–57] to provide mobility support. As a signaling protocol, SIP is widely used to establish, modify and terminate sessions. Figure 2.3 presents a typical procedure on how TMSP operate with the help of SIP. If CN wants to establish a connection with its peer MN, it first sends an invitation message to the SIP proxy server which is responsible for forwarding the signaling message to MN. If MN accepts the invitation, it will reply with an OK response to CN. Finally, CN sends an acknowledgement message to MN directly to

Figure 2.3: Terminal Mobility Support Protocol

confirm the establishment of the connection. During the SIP session, if MN moves to another network and acquires a new address, the SIP will help MN register the new address in its SIP registrar and re-invite the CN. To hide the application from the address change, TMSP maintains a binding in each communication entity. The binding maps a peer's IP address used for session establishment to the IP address currently used by the peer. In Figure 2.3, when CN sends a packet to MN which has moved from access network (AN1) to AN2, the TMSP module in CN will replace the packet header with MN's address in AN2. After the packet is received successfully by MN, the TMSP module will restore the packet header with the original IP address before handing the packet to the application. Through the IP header swapping, TMSP solve the mobility issue by letting the original IP address play the identification role and latest IP address play the location role.

Through decoupling the two conflicting roles bound to the IP address, these global mobility protocols [58–60] (e.g., MIP, HIP and TMSP) allow MN to maintain its reachability

no matter where it moves. However, global mobility protocols do not work well and they suffer a lot of disadvantages [61]. For example, the global mobility requires MN to update its location with HA or CN every time it changes its attaching network. If MN roams away from its HA or CN, the update operation will take a long time and the quality of service will be reduced. Moreover, for some global mobility protocols (e.g., MIP) which rely on HA to redirect traffic, the user data will suffer from triangular routing. Traffic is not sent directly between CN and MN. Instead, the traffic is firstly sent to HA. When HA receives traffic on behalf of MN, the traffic will be encapsulated and redirected to MN at the new location. The sub-optimal routing introduces transmission latency to the global mobility protocols. Moreover, the new address contained in the update message will expose MN's topological location to CN and potential eavesdroppers. The exposure will render MN to be easily tracked [62].

2.1.2 Local Mobility

In respect to the disadvantages of global mobility, especially for the longer update latency, local mobility has been developed. Unlike global mobility which enables users to move universally, local mobility restricts the service within a limited area. The limited area is named local domain. Figure 2.4 presents the typical topology of a local domain. A local domain usually has four kinds of entities: Domain Gateway Router (DGR), Access Router (AR), Access Point (AP) and Mobile Node (MN). DGR is the topological anchor point of the local domain. All traffic in and out of the domain will go through the DGR. At the same time, DGR is the converging point for multiple Access Networks (AN). In Figure 2.4, both the WiFi network and UMTS network connect to DGR. Although different wireless technologies have different access network topologies (e.g., the AN1 of WiFi is different from AN2 of UMTS), they can all be simplified into a model comprising AR and AP (e.g., AN3 in Figure 2.4). AR is the first hop router which is in charge of an access network. Inside AN, data sent from user device is managed by wireless technology protocol. Outside AN,

Figure 2.4: The Topology of Local Domain

the data is routed following the IP routing rule. AP is the base station that provides wireless link to users and MN is the device held by users such as PDA and smartphones.

Restricting mobility service in a limited area can help local mobility improve the performance of global mobility. As we know, the handover delay of global mobility is mainly caused by the long distance between MN and HA/CN. To reduce the distance, local mobility designates a mobility agent (usually played by DGR which is the converging node of the domain) in the domain to perform the location tracking and packet routing for HA/CN. When

MN moves in the local domain, the mobility agent will receive traffic on behalf of MN and then the traffic will be redirected by the mobility agent to the wireless link to which MN is currently attached to. As long as MN roams in the local domain, the HA/CN is kept away from the location update. If MN moves from one AN to another, the update message will be sent to the mobility agent instead of HA/CN. The shorter distance between MN and mobility agent will reduce the handover latency.

There is a variety of local mobility protocols [63–65] that provide mobility support to MN in a local domain, such as Cellular IP (CIP) [66], HAWAII [67], Hierarchical Mobile IPv6 (HMIPv6) [68] and Proxy Mobile IPv6 (PMIPv6) [69]. These mobility protocols vary in terms of traffic overhead, handoff management, paging support and intra-domain traffic routing. In the following part of this section, we will first describe the workflow of each local mobility protocol and then we summarize their difference in a table.

Cellular IP (CIP)

Cellular IP (CIP) [66] is a local mobility protocol that integrates location management and routing support. Figure 2.5 shows an overview of CIP. Under CIP, the routers in a local domain form a tree hierarchical topology. The root of the tree is DGR of the local domain. If MN changes its attaching network (e.g., move from network C to network D), it notifies the newly attached router with an update message. The update message can either be a data packet or a signaling packet. When MN is in idle status, it delivers the notification with a signaling packet; otherwise, MN can update the routers by replying with a data packet. No matter what the update message is, it will be delivered back to DGR hop-by-hop. Each cross-router along the path from MN to DGR will interpret the update message and changes its routing table if necessary. For example, on receipt of an update message that indicates MN has moved to network D, router A in Figure 2.5 will change the host routing for MN. When router A receives a packet destined at MN, it will redirect the packet to router B instead

Figure 2.5: Cellular IP

of router C.

Through the hop-by-hop redirection, CIP can deliver traffic to MN successfully. However, CIP has many disadvantages. Firstly, the strict tree-hierarchical topology of CIP limits its deployment. Since CIP requires the update message to be routed back to DGR along the same path as the traffic delivered to MN, the CIP domain does not support non-tree topology. Moreover, all routers in CIP domain must handle the Cellular IP update message. This increases the upgrade cost for the local domain. In addition, CIP does not support intra-domain traffic route optimization (RO). If a MN sends a packet to its corresponding node

(CN) which is connected to the same domain, the packet must first be transferred to the DGR which will subsequently deliver it to the destination. The suboptimal routing issue degrades the performance of intra-traffic and lowers the utilization of network resource.

HAWAII

HAWAII [67] is another widely deployed local mobility protocol. Figure 2.6 shows its workflow. The topology of HAWAII domain is similar to CIP which is organized as a hierarchical tree. Each router in the topology tree is expected to interpret the signaling message. When MN changes the attaching network in the local domain (e.g., from network R4 to R5), it first sends an update message to R4 which is attached to by MN previously. On receipt of the update message, R4 knows the new network where MN is residing. Then, R4 changes the host routing for MN. If there are data packets arriving at R4, R4 will tunnel the packets to the new location of MN which is R5 in Figure 2.6. At the same time, to ensure successful delivery to MN's new location, R4 replies a signaling message to MN. The signaling message will be interpreted hop-by-hop by the routers along the path to MN. If the routers find that the next hop to MN's new location is different, they will change their host routing for MN correspondingly. After the update operation is completed, MN can continue receiving its traffic in the new location.

Unlike CIP which sends the update message hop-by-hop back to DGR, HAWAII requires MN to first send an update message to the old AP. Then, the old AP will buffer and redirect the packets to MN's new location. The buffering and forwarding behavior will cause HAWAII less packet loss during handover [70]. However, the HAWAII also suffer some disadvantages such as packet reordering during the handover, the strict requirement on tree-hierarchical topology and the suboptimal routing for intra-domain traffic.

Figure 2.6: HAWAII

Hierarchical Mobile IPv6 (HMIPv6)

Hierarchical Mobile IPv6 (HMIPv6) [68] is another widely deployed local mobility protocol. It reuses the signaling module of Mobile IP and extends it for the local domain. Figure 2.7 presents the workflow of HMIPv6. When MN enters the local domain, it first selects one mobility anchor point (MAP) from the available MAP list. The selected MAP plays the role of mobility agent for MN. If MN changes the attaching network in the domain (e.g., moving from AN1 to AN3 in Figure 2.7), the serving MAP will track the movement and redirect the traffic to the new location. HMIPv6 allocates two addresses to MN. One is Regional Care-of-Address (RCoA) and the other is On-link Care-of-Address (LCoA). RCoA is the address indicating which domain MN is residing right now. After one MAP is designated as

Figure 2.7: Hierarchical Mobile IPv6

the serving mobility agent, it assigns an RCoA to MN from its address pool. MN registers the RCoA to its HA and binds RCoA with the home address. Since RCoA does not change as long as MN stays in the domain, HMIPv6 can keep HA agnostic of the local movement. However, the RCoA is not enough to locate MN and HMIPv6 assigns an LCoA to MN. The LCoA is the address indicating which wireless link MN is attached to. Every time MN moves to a new network, the LCoA is updated and MN sends the latest LCoA to MAP to bind with its RCoA.

With RCoA and LCoA, HMIPv6 divides the packet forwarding into two parts. When HA receives a packet destined to MN, HA encapsulates the packet and tunnels it to MAP. After MAP receives the packet and finds the packet is sent to RCoA, it encapsulates the packet with MN's LCoA and sends it out. Since LCoA is topologically correct with MN's

current location, the packet will be received by MN successfully. Moreover, because MAP has a much shorter distance from MN, HMIPv6 can reduce the mobility latency when MN handovers within the domain.

Proxy Mobile IPv6 (PMIPv6)

PMIPv6 is a network-based mobility protocol [61, 71, 72] which allows MN to roam in the local domain without any involvement of protocol signaling. To implement it, PMIPv6 designates two new network entities - Mobile Access Gateway (MAG) and Local Mobility Anchor (LMA) - to perform the mobility signaling on behalf of MN. MAG is an access router and it provides wireless link to MN. LMA is the topological anchor point of the domain and it maintains the reachability status for MN. Figure 2.8 shows the workflow of PMIPv6. When MN moves from MAG1 to MAG2, both of which are under the administration of LMA, it will first send a router solicitation (RS) message to MAG2. The RS message is used to initiate the Router Advertisement (RA) message from MAG. When MAG2 receives the RS message, it senses that MN has moved into the network. Hence, MAG2 sends a Proxy Binding Update (PBU) message to LMA on behalf of MN. The PBU contains MN's address used in previous MAG network. If MN has passed the authentication procedure, LMA will reply with a Proxy Binding Acknowledgement (PBA) message to MAG, carrying MN's address configuration. With the PBA message, MAG will construct a Router Advertisement (RA) message and send to MN via unicast. Since the address configuration in MAG2 is the same as MAG1, MN can sense the movement and hence will not participate in the mobility procedure.

To deliver traffic to MN, LMA builds a bi-directional tunnel with MAG whenever MN changes its location. Since LMA is the topological anchor point of the domain, LMA will receive data packets before they arrive at MN. When LMA receives the data packets, it will encapsulate the packets and send them to MAG. Then, MAG will decapsulate the packets

Figure 2.8: Proxy Mobile IPv6

and deliver them to MN. If MN wants to send out data packets to its communicating peer, MAG will first tunnel the data packets to LMA. Then, LMA will decapsulate the packets and send them out according to their inner IP header.

2.1.3 Comparison of Different Local Mobility Protocols

Although multiple local mobility protocols have been proposed as the solutions for the handover within a local domain, they do not perform well. In [58, 73–76], authors have listed the disadvantages of the existing local mobility management protocols and they are summarized in Table 2.1.

From Table 2.1, we can see that although four local mobility protocols provide the mo-

Table 2.1: Comparison of Different Local Mobility Protocols

Local Mobility Protocol	Cellular IP	HAWAII	HMIPv6	PMIPv6
Network Topology	strict	strict	loosen	loosen
Intra-domain RO	No	No	No	No
Entities Involved	all routers and MN	all routers and MN	DGR and MN	DGR, AP and MN
Traffic delivery	host routing	host routing	tunnel to MN	tunnel to AP
Traffic Overhead	0	0	tunnel header from DGR to MN	tunnel header from DGR to AP

bility support for MN in the local domain, they do not provide routing optimization (RO) service for the intra-domain traffic. If MN wants to send a packet to its communicating peer which also resides in the same domain, the packet should be delivered to DGR first and then DGR will send the packet to MN's peer. Nowadays multimedia applications such as Face Time and online games are popular, the sub-optimal routing described above wastes bandwidth resource and lowers the QoS. Besides, CIP and HAWAII impose a strict hierarchical tree requirement on the domain topology. The strict requirement increases the implementation cost to upgrade the current domain to local-mobility-enabled domain. Furthermore, because HMIPv6 and PMIPv6 employ IP tunnel technology to deliver packets within the local domain, they suffer from higher the traffic overhead and bandwidth utilization ratio is thus reduced.

Aiming at improving the performance of existing local mobility protocols, a new framework for local mobility - Efficient Framework for Local Mobility (EFLoM) is proposed. The first aim of EFLoM is to optimize the routing path for intra-domain traffic. Under EFLoM, the intra-domain traffic is no longer delivered to DGR. Instead, it follows the shortest

path between the communication peers. EFLoM also aims to reduce the traffic overhead, especially for the wireless link. Since current wireless application always suffers from the limited bandwidth, the higher utilization of the wireless link can improve the overall network performance. In addition, EFLoM aims to adapt to loosen local domain which can reduce the implementation cost. EFLoM employs three network entities: Local Anchor Router (LAR), Wireless Access Gateway (WAG) and Mobile Node (MN). With the help of the three network entities, EFLoM can reduce the handover delay for MN. Moreover, EFLoM uses an IP header swapping mechanism to avoid the traffic overhead and improve the network's throughput. To evaluate the performance of EFLoM, several experiments are carried out to compare EFLoM with Hierarchical Mobile IPv6 (HMIPv6) [68] and Proxy Mobile IPv6 (P-MIPv6) [69] - both of which are common local mobility management protocols - in terms of handover delay, throughput and signaling cost. Analysis and simulation results show good performance for EFLoM.

2.2 Inter-Domain Mobility Protocol

With the popularity of the localized domains, many users move across the domains. Hence, inter-domain mobility is needed to enlarge the movement of users as well as provide improved mobility service to users. For instance, a user starts an Internet telephony or real video session with his friend on NTU campus. Then, he decides to move to NUS which is far away from NTU. On the way from NTU to NUS, the user may move across different local domains and he expects to keep his application uninterrupted. An inter-domain mobility protocol can help solve this issue and provide users a transparent mobility service when they are roaming across neighboring domains.

Before embarking on inter-domain proposal, three new terms: home local mobility anchor (HLMA), previous foreign local mobility anchor (PFLMA) and new foreign local mo-

bility anchor (NFLMA) are introduced. HLMA is the registering domain of MN. When MN first enters a series of domains, it acquires its address configuration in the HLMA domain. Then, MN registers itself with HLMA and starts to be serviced in the HLMA domain. If MN moves from one PMIPv6 domain into another, the previously residing domain is MN's PFLMA domain and the newly visited domain is MN's NFLMA domain.

2.2.1 Neumann's Proposal

Neumann [77,78], shown in Figure 2.9, is an inter-domain protocol which provides mobility support across neighboring domains. It is based on PMIPv6 protocol and extends its support across domains. When MN attaches to NFLMA's wireless link, MAG will send a register Proxy Binding Update (PBU) message to NFLMA on behalf of MN. Upon receiving the PBU message, NFLMA will send a PBU to the HLMA to query MN's address configurations. After getting the answer from HLMA, NFLMA will reply with a Proxy Binding Acknowledgement (PBA) to MAG, carrying MN's address configuration. As PBA does not change in the NFLMA domain, MN's address will remain the same as the one in the HLMA domain. It is noted that if MN returns to its HLMA domain, there is no query operation and HLMA will reply with a PBA to MAG directly.

After the signaling procedure is completed, MN can begin to receive traffic in the NFLMA domain. Under Neumann's proposal, the packet transmission is divided into two steps. Firstly, the packet is delivered from HLMA to NFLMA. Then, NFLMA delivers the packet to MN within its domain. According to Neumann's signaling, MN's IP address can be kept unchanged during the movement. Thus, HLMA continuously serves as MN's topological point and the HLMA will receive the traffic on behalf of MN. When HLMA receives a packet destined to MN, it will check MN's location and tunnel the packet to NFLMA. After NFLMA unpacks the packet, NFLMA will redirect the packet to MAG whose link is

Figure 2.9: The Architecture of Neumann's Proposal

currently attached to by MN and it will forward the packet to MN.

Although Neumann's proposal can provide cross-domain mobility support to MN, there are some problems. First of all, in Neumann's proposal, LMA plays both HLMA role and NFLMA role. Consequently, LMA has to keep Binding Cache Entry (BCE) for two categories of MN. The first is the MN who registers itself in this domain. As MN's HLMA, LMA has to keep the BCE for MN no matter which domain MN resides in. In addition, LMA also needs to keep the BCE for MN who is currently visiting its domain. Hence, under Neumann's

proposal, the number of BCE is increased. If there are many MN visiting the domain, the number of BCE will become a burden for LMA and will limit the serving range of LMA. The second issue is the triangular routing problem. The triangular routing problem is similar with MIP problem analyzed in [79]. In Neumann's proposal, the data packet is transmitted along the path from HLMA to NFLMA, and finally to MAG. When MN moves to FLMA's domain which is far from HLMA, the distance between HLMA to FLMA will render the traffic to incur a greater transmission delay. In order to overcome these disadvantages, an alternative solution to address the inter-domain handover challenges is proposed.

2.3 Multi-Path Scheduling

The second issue in the FWN integration is the bandwidth aggregation. Current wireless applications often suffer from the limited bandwidth of wireless links. With the development of different wireless technologies such as WiMAX, WiFi and UMTS, the future wireless local domain is believed to be a place where these wireless technologies overlap one another, compete and complement one another. In such environment, users can improve the experience of wireless usage by accessing multiple wireless technologies. As we know, wireless link is easily affected by several adverse effects such as radio propagation loss, shadowing and multipath fading. When users access one single wireless link, they may not experience a stable performance. However, if users access multiple networks simultaneously, they can utilize the performance of different links to alleviate the effects of bandwidth fluctuation in a single wireless link. Moreover, the multiple access can aggregate the bandwidth to support bandwidth-hungry applications such as live video streaming, or reduce the communication time for highly mobile users whose connection time is thus limited.

Before users enjoy the advantages of multiple-access technologies, a suitable multipath scheduling algorithm is needed. Multipath scheduling is always a hot research topic and

there are a lot of multipath scheduling algorithms. They can be classified into two broad categories according to their different implementation strategies. One is end-to-end scheduling and the other is the network proxy scheduling. The end-to-end scheduling aggregates the bandwidth of multiple paths by changing the implementation of the transport protocols at both communication peers. The changes can help traffic adapt the multipath routing and reduce the disorder packets. The network proxy scheduling, on the other hand, is a kind of multipath scheduling which is implemented at the intermediate network device such as bifurcating routers. When data traffic arrives, the routers split it and allocate it to multiple delivery paths. The splitting and allocating behavior can help network proxy scheduling use multiple bandwidth resource simultaneously without any requirement on upgrading the sender or receiving devices.

2.3.1 End-to-End Multipath Scheduling

When splitting one flow to multiple paths, designers have to take care of the phenomenon of packet reordering. Packet reordering is resulted from the difference in transmission delay. The earlier sent packet may be overtaken by its successive packets because it is delivered along a slower path. The disordered arrival of the packets will be misinterpreted as an indication of packet loss and can cause the sender to downgrade the performance of the TCP traffic inappropriately. To reduce the packet reordering, end-to-end multipath scheduling revises the implementation in the transport protocol for both the sender and the receiver. The modification can compensate the delay difference between multiple paths and ensure an in-order delivery to the receiver's application. For example, the Weighted Round Robin-Jump Ahead (WRR-JA) [80] tries to reduce the out-of-order packets by adjusting the sending sequence at the sender.

Figure 2.10 presents a logical topology of end-to-end multipath scheduling. In the topol-

Figure 2.10: End-to-End Multipath Scheduling

ogy, there are several paths between the sender and the receiver. Each path can achieve an in-order delivery with a different transmission delay. However, the delay difference between these paths may result in disorder reception. In the following, three scheduling mechanisms aiming at balancing the delay difference will be introduced.

Weighted Round Robin-Jump Ahead (WRR-JA)

Weighted Round Robin-Jump Ahead (WRR-JA) [80] is a multipath scheduling scheme designed to reduce the occurrence of out-of-order packets by transmitting the packets in a “disordered” fashion at the sender side. To compensate for the longer delay in the slower path, WRR-JA reduces the packet’s queuing time by sending packets meant for the slower path ahead of those scheduled for the faster paths in the queue. Although the packets suffer longer transmission delay in the slower path, their queuing delay is reduced. Hence, the packets of the slower path can arrive before those delivered via the faster path, resulting in

ordered packet reception.

Concurrent Multipath Transfer (CMT)

Concurrent multipath transfer (CMT) [81, 82] is a multipath scheduling which employs the Stream Control Transmission Protocol (SCTP) to split traffic across multiple paths. Through identifying three negative side-effects with the multipath scheduling, which are: (1) the unnecessary retransmission mechanism; (2) inappropriate growth of the congestion window for TCP flow; (3) the increase of ACK resulting from delayed ACK, CMT proposes three algorithms to solve them.

To prevent the unnecessary fast retransmission, CMT develops a split fast retransmission algorithm which establishes virtual retransmission queue for each path. The new retransmission enables CMT to apply the correct fast retransmission procedure on a per-destination basis. To avoid reduction in congestion window, CMT proposes a new congestion window to track the earliest outstanding sequence number per destination and update the congestion window size accordingly. To reduce the increase of ACK, CMT proposes that the receiver report the packet number received between neighboring ACK. With this number, the sender can infer the number of missing reports and then reduce the out-of-order packets.

Multipath Transmission Control Scheme (MTCS)

Multipath Transmission Control Scheme (MTCS) [83] is another end-to-end multipath scheduling scheme which is developed to support split video traffic over parallel paths. To balance the different conditions along each path, MTCS develops a load sharing module to handle the congestion loss and transmission loss. The load module can help MTCS find the optimal transmission throughput. At the same time, MTCS implements a feedback control mechanism to maximize the successfully delivered data to receiver. With MTCS, users

are expected not only to enjoy improvement on the aggregated throughput but also reduce out-of-order packets.

Of course, the end-to-end multipath scheduling is not limited to the three implementations illustrated above. There are others such as Reliable Multiplexing Transport Protocol (R-MTP) [84], Parallel Transmission Control Protocol (pTCP) [85] and Load Sharing - Stream Control Transmission Protocol (LS-SCTP) [86], all of which are based on changing the protocol stack in both the sender and the receiver.

2.3.2 Network Proxy Multipath Scheduling

Although end-to-end scheduling provides a solution for aggregating bandwidth of different paths, they are cumbersome to implement as they involve many changes in the protocol stack in communication peers. The changes pose difficulty and increase the cost to deploy them, especially for mobile devices which have resource constraints. To mitigate the complication of modifying the mobile devices, the network proxy scheduling is proposed. By implementing the scheduling at the intermediate node, network proxy scheduling has the advantage of being transparent to mobile users and involve only minimal changes. Compared with end-to-end scheduling, the network proxy solution is easy to deploy and beneficial for legacy applications.

Figure 2.11 presents the application scenario of network proxy scheduling. First, the traffic is delivered from the sender to the proxy splitter which is normally assumed by the bifurcating routers in the middle of the transmission path. After the traffic is received, packets are split and delivered along multiple paths simultaneously. The paths between the splitter and the receiver may differ in many aspects such as path delay and capacity. The network proxy scheduling will take the difference into account and aggregate their throughput effectively.

Figure 2.11: Network Proxy Multipath Scheduling

Opportunistic Multipath Scheduling (OMS)

One example of network proxy scheduling is Opportunistic Multipath Scheduling (OMS) [87, 88]. Unlike WRR-JA, OMS scheme is deployed in the intermediate routers at the ISP level instead of at the source node. Hence, OMS does not require any change to the sending node and thus it can be deployed easily. In order to cope with the time-varying characteristic of the wireless channel, OMS develops an opportunistic scheduling technique to distribute packets across different paths.

The scheduling procedure is shown in Figure 2.12. Before scheduling, OMS employs a probing message to estimate the transmission delay for each path. When a data packet arrives at the splitter, OMS selects the path with low delay and high quality among all candidate paths. The selection is based on the calculation of opportunistic values, one for each path. Each opportunistic value reflects the condition of the path and a lower value indicates that the path is less congested. At the same time, to ensure the number of packets scheduled on each path complies with the desired load ratio specified by the users in the long run, a weighting factor $V = (v_1, \dots, v_N)$ is further considered in the calculation of the opportunistic values, where $v_i, i = 1, \dots, N$, specifies the weighting for path i . Each time after OMS distributes

Figure 2.12: Opportunistic Multipath Scheduling

the packet over the selected path, e.g., path i , the weight v_i will be updated.

Flowlet Aware Routing Engine (FLARE)

Flowlet Aware Routing Engine (FLARE) [89,90] mainly performs scheduling for TCP flows which suffer out-of-order issue. Through investigation, FLARE finds that TCP flow comprises several flowlets, each of which is a group of packets. The packet interval in the group as shown in Figure 2.13, is much less than others. In Figure 2.13, the TCP flow is divided into three flowlets. The gap between flowlet is much wider than the gap between packets within the flowlet. The flowlet comes from the burst characteristic of TCP flows. When TCP source transmits traffic, the amount of traffic sent to the receiver is constrained by the size of the congestion window. If the number of sent packets reaches the size of the congestion window, the TCP source will cease transmitting until it receives an acknowledgement packet from the destination node.

The large flowlet gap provides a good opportunity to compensate for the delay disparity between the different paths. If FLARE switches delivery path at the flowlet interval, it

Figure 2.13: Flowlet Aware Routing Engine

can reduce the possibility of out-of-order packets. When a packet arrives at the scheduler, FLARE judges whether the packet is the head of a flowlet by monitoring the packet interval. If the interval is much wider than the others, the packet is at the beginning of the flowlet. When a new flowlet arrives, FLARE splitter chooses the path whose load ratio is furthest away from its desired share and distributes flowlet to the chosen path. For the packets within the flowlet, FLARE does not perform path switching and delivers them along the same path.

Through splitting flow at the granularity of burst, FLARE can reduce the occurrence of out-of-order packets as well as achieve good load balancing across multiple paths. However, FLARE does not always perform well. First of all, the flowlet interval does not exist all

the time. The second disadvantage for FLARE is that FLARE does not consider packet loss during the scheduling. As TCP flow is a guaranteed delivery service, it retransmits the packet when packet loss occurs. The retransmission cost is high in FLARE because the path congestion can cause the loss of a burst of packets.

As network proxy scheduling can free mobile users from the resource constraints and achieve bandwidth aggregation in a less costly manner, the proposed scheduler will be implemented in the proxy splitter. Taking the typical topology of FWN into consideration, the DGR is a good entity to install the scheduling algorithm. The proposed algorithm named Adaptive Load Balancing Algorithm (ALBAM), is based on the concept of flowlet in TCP flow. In order to address the two disadvantages of FLARE, ALBAM develops two algorithms: Virtual Flowlet Division (VFD) and Packet Number Estimation Algorithm (PNEA). VFD is a traffic division algorithm. It is used to group several adjacent packets together when flowlets are missing. In the stable phase of a TCP flow, VFD can help ALBAM divide traffic into several groups and schedule these groups across multiple paths. PNEA is a technique used to estimate the number of on-delivery packets in each path. The estimation can help ALBAM avoid sending packets to congested path and prevent packet retransmission during the scheduling. With these techniques, ALBAM can improve the performance of TCP flows when splitting them to multiple paths.

Part I

Intra-Domain Handover

Chapter 3

EFLoM: An Efficient Framework for Local Mobility

In this chapter, a new framework: Efficient Framework for Local Mobility (EFLoM) is proposed. EFLoM is developed to handle the issue of local mobility management. To enable MN to move across networks within one domain, EFLoM employs three network entities: Local Anchor Router (LAR), Wireless Access Gateway (WAG) and Mobile Node (MN). With the help of the three network entities, EFLoM can reduce the handover delay for MN. Moreover, EFLoM uses an IP header swapping mechanism to avoid the traffic overhead and improve the network's throughput. To evaluate the performance of EFLoM, several experiments are carried out to compare EFLoM with Hierarchical Mobile IPv6 (HMIPv6) [68] and Proxy Mobile IPv6 (PMIPv6) [69] - both of which are common local mobility management protocols - in terms of handover delay, throughput and signaling cost. Analysis and simulation results show good performance for EFLoM.

3.1 Introduction

The recent development of mobility protocols has yielded a two-layer hierarchical mobility framework. In the framework, users' movement is classified into two types: the movement within a local domain and the movement outside a local domain. For the two kinds of movement, there are correspondingly two mobility protocols. Local mobility is responsible for the handover within the local domain whereas global mobility is responsible for the movement outside the local domain. When users roam within a local domain, they can employ local mobility to maintain their reachability. If users move out of the coverage of local domain, the global mobility can help them handle the session handover issue.

Actually, local mobility is extended from global mobility and they aim to improve the performance of global mobility in the local domain. Although there are a lot of local mobility protocols such as Cellular IP (CIP) [66], HAWAII [67], Hierarchical Mobile IPv6 (HMIPv6) [68] and Proxy Mobile IPv6 (PMIPv6) [69], they do not perform very well. Firstly, they do not provide routing optimization (RO) service for the intra-domain traffic. If MN wants to send a packet to its communicating peer which is also resides in the same domain, the packet will be delivered to DGR which will then send it to MN's peer. In these days where multimedia applications such as Face Time and online games are popular, the sub-optimal routing will waste bandwidth resource and lower the QoS. Secondly, CIP and HAWAII impose a strict hierarchical tree requirement on the domain topology. The strict requirements increase the implementation cost needed to upgrade the current domain to a local-mobility-enabled domain. Thirdly, because HMIPv6 and PMIPv6 both employ IP tunnel technology to deliver packets within the local domain, they suffer from traffic overhead and thus bandwidth utilization ratio is reduced.

Aiming at improving the performance of existing local mobility protocols, a new framework for local mobility - Efficient Framework for Local Mobility (EFLoM) is proposed.

The first aim of EFLoM is to optimize the routing path for the intra-domain traffic. Under EFLoM, the intra-domain traffic is no longer delivered to DGR. Instead, it follows the shortest path between the communicating peers. Then, EFLoM aims to reduce the traffic overhead, especially for the wireless link. Since current wireless application always suffers from the limited bandwidth, the higher utilization of wireless link can improve the overall network performance. In addition, EFLoM aims to adapt to loosen local domain which can reduce the implementation cost. EFLoM employs three network entities: Local Anchor Router (LAR), Wireless Access Gateway (WAG) and Mobile Node (MN). With the help of the three network entities, EFLoM can reduce the handover delay for MN. Moreover, EFLoM uses an IP header swapping mechanism to avoid the traffic overhead and improve the network's throughput. Comparative evaluation through extensive simulation shows that EFLoM achieves good performance over HMIPv6 and PMIPv6.

Among the local mobility protocols, HMIPv6 and PMIPv6 are chosen for comparison with EFLoM. Firstly, both HMIPv6 and PMIPv6 are the most cited local mobility protocols. Secondly, both HMIPv6 and PMIPv6 are based on a loose topology tree which does not limit the application. Other local mobility protocols such as Cellular IP and HAWAII require a strict hierarchical tree on the domain topology and it increases the cost to upgrade the current system to a domain-mobility-enabled system.

The remainder of this chapter is organized as follows. Section 3.2 gives a brief introduction on the related works and the motivations of the proposed EFLoM. Section 3.3 gives a detailed illustration of EFLoM which comprises three subsections. Section 3.3.1 presents the system architecture and EFLoM's three network entities. Section 3.3.2 gives a description of the handover procedure of EFLoM and Section 3.3.3 briefs the forwarding procedure of EFLoM. Section 3.4 presents the qualitative comparison of EFLoM with HMIPv6 and PMIPv6, both of which have similar topology requirement as EFLoM. Section 3.5 presents the analysis and simulation results of EFLoM. Section 3.6 concludes the chapter.

3.2 Motivation and Related Work

3.2.1 Related Work

Before introducing the proposed EFLoM, an introduction on Hierarchical Mobile IPv6 (HMIPv6) [68] and Proxy Mobile IPv6 (PMIPv6) [61, 69, 71, 72] is given. The reason why HMIPv6 and PMIPv6 are used for comparison is that the two protocols can be applied in a loose domain topology which is similar to EFLoM. The other two common local mobility protocols, Cellular IP (CIP) [66] and HAWAII [67] impose a hierarchical tree on the network topology. The details of four protocols can be referred to in Section 2.1.2.

3.2.2 Motivations and Goals

Although HMIPv6 and PMIPv6 can handover the session within a local domain, our observations show that they suffer some problems. Firstly, both HMIPv6 and PMIPv6 do not optimize the delivery path of the intra-domain traffic. Under HMIPv6 and PMIPv6, all the intra-domain traffic is passed through the gateway router instead of passing directly to the receiver. The triangular routing issue degrades the transmission performance. The second problem with HMIPv6 and PMIPv6 is their traffic overhead. Since both HMIPv6 and PMIPv6 employ tunneling technology to deliver traffic within the domain, the extra tunnel header introduced in every data packet will consume network bandwidth. Although PMIPv6 removes the packets' tunnel header on the wireless link, the traffic overhead on the wired link still exists.

In order to improve the performance of HMIPv6 and PMIPv6, a new protocol, Efficient Framework for Local Mobility (EFLoM) is proposed. Unlike HMIPv6 and PMIPv6, EFLoM employs an IP swapping mechanism [52, 91, 92] to deliver traffic. IP swapping helps EFLoM to remove the tunnel header when delivering traffic. When EFLoM receives a data package,

it updates the destination IP of the data package with the receiving node's latest address. When the data package arrives, the receiving node restores the destination IP address to keep the IP change hidden from the applications. Comparing with other mobility protocols, EFLoM has the following advantages:

1. **Traffic Overhead** – The extra packet header will be removed from both wireless and wired links. EFLoM will introduce zero traffic overhead in the local domain. Hence, the throughput of the local domain will be maximized.
2. **Delivery Optimization** – The Ping-pong effect is removed. In EFLoM domain, packets will be delivered to the destination along an optimal path.
3. **Protocol Signaling Cost** – EFLoM will reduce MN's signaling cost and save on MN's power consumption

3.3 Efficient Framework for Local Mobility

3.3.1 System Architecture and Network Entities

To achieve a better performance over HMIPv6 and PMIPv6, EFLoM employs three kinds of network entities: Local Anchor Router (LAR), Wireless Access Gateway (WAG) and Mobile Node (MN). Figure 3.1 presents the system architecture of EFLoM and it is similar to that of HMIPv6 and PMIPv6.

LAR is the gateway router of the local domain. It acts as the proxy HA for MN. When MN roams in the local domain, LAR will receive traffic on behalf of MN and redirect the traffic to MN's residing link. To attain this, EFLoM assigns two addresses to MN: Regional Care-of-Address (RCoA) and Link Care-of-Address (LCoA). The RCoA is an address obtained by MN when MN first enters the local domain. It acts as the identification for MN

in the local domain. MN uses the RCoA as the location indication to update its HA [93] or communicating peers [50]. When MN moves within the local domain, the RCoA does not change. Hence, MN does not need to send binding update message to the HA or communicating peers unless it moves out of the local domain. However, the RCoA is not topologically correct with MN's residing link. To locate MN's position, EFLoM assigns an LCoA to MN. The LCoA is an address consistent with MN's location and it is updated whenever MN changes the attaching point. At the same time, LAR will keep both RCoA and LCoA for MN. If LAR receives traffic destined at MN, the LAR will change the packet's destination from RCoA to LCoA. Then LAR will send out the updated packet. Since the LCoA is topologically correct with MN's location, MN can receive the packet successfully. Through binding the RCoA with the LCoA, EFLoM can redirect the traffic within its domain.

Figure 3.1: System Architecture of EFLoM

However, LAR alone cannot handle the local mobility. EFLoM needs WAG to manage the handover for MN. WAG is an access router which is in charge of a wireless network in the

EFLoM domain. It corresponds to the AR entity in the local domain. It interconnects several AP which provide wireless link to MN. When MN moves to the WAG's network, WAG will send a Register Request (RR) message to LAR for the incoming MN. If the MN is an authenticated user, WAG will provide access service to MN. For example, WAG will assign a new LCoA to MN and deliver MN's traffic to the corresponding wireless link. Moreover, if MN wants to send a packet to its peer which also resides in the EFLoM domain, WAG will update the packet's destination address and ensure the packet is routed via the optimal path to MN's peer. The later section will cover the procedure of forwarding a packet in details.

The third entity of EFLoM is MN. The MN is a wireless device that roams within the local domain. Under EFLoM, every MN has an IP switching agent. This agent is used to process the IP header of MN's traffic. Figure 3.1 presents the protocol model of MN. The IP switching agent lies between Network layer and Data Link layer. When MN receives a packet destined at MN's LCoA, the IP swapping agent will restore the packet's destination to RCoA and then pass the packet to the network layer. At the same time, if the network layer wishes to send out a packet, the IP swapping agent will update the packet's source address with MN's LCoA. Through managing the IP header in the IP switching agent, MN can keep its communication session unaware of the change of LCoA. Hence, MN can continue its existing connection even if it moves to another network in the domain. For detail operation of the IP swapping mechanism, please refer to Section III of [52].

3.3.2 Registration Procedure

Before performing registration procedure, MN needs to acquire a regional care-of-address (RCoA). The RCoA indicates which domain MN is residing. When MN enters a new domain, it first employs stateless configuration or DHCP [94] to obtain its RCoA. Then, MN registers the RCoA with its global mobility agent (e.g., under MIP, MN binds the RCoA with

its home address and sends them to the home agent). Since RCoA does not change as long as MN resides in the domain, the global mobility protocol is kept agnostic of the movement within the domain.

After acquiring RCoA, MN begins to perform registration procedure which is presented in Figure 3.2. When a handover happens, the newly attached WAG sends out a Router Advertisement (RtrAdv) message to MN. The RtrAdv message helps MN obtain a LCoA in the new network (via stateless configuration or DHCP). Then, WAG binds the LCoA with RCoA which is acquired by MN on entry into the local domain and sends them to LAR in a local binding update (LBU) message. If LAR accepts the LBU message, it sets up a binding entry between RCoA and LCoA. The binding entry will be used to forward traffic. At the same time, LAR replies with a local binding acknowledgement (LBA) message to WAG. Upon receiving the LBA, the registration procedure is completed and MN can begin packet transmission in the new location.

3.3.3 Packet Forwarding

After the registration procedure, EFLoM can begin packet forwarding. Within the EFLoM domain, the traffic is classified into two types: the inter-domain traffic and the intra-domain traffic. If corresponding node (CN) and MN reside in the same local domain, their traffic is intra-domain traffic; otherwise, it is inter-domain traffic. For the two kinds of traffic, EFLoM employs two different forwarding procedures.

Figure 3.3 presents the path of inter-domain traffic in EFLoM. When CN sends out a packet to MN, the packet will first arrive at LAR with RCoA as the destination address. Then, LAR will interpret the packet and update the packet's destination address with MN's LCoA. Since LCoA is topologically correct based on MN's current location, the packet will arrive at MN after being updated by LAR. MN's IP swapping module will restore the packet's

Rtr Adv: Router Advertisement
 Rtr Sol: Router Solicitation
 LBU: Local Binding Update
 LBA: Local Binding Acknowledgement

Figure 3.2: The Handover Procedure of EFLoM

destination address with MN's RCoA and send it to the applications in the upper layers.

If MN wants to send a packet to CN, MN's upper layer application will use the RCoA as the packet's source address. Before the packet goes on the wireless link, MN's IP swapping module will update the packet's source address with MN's LCoA, and sends it out. As CN resides outside the local domain, the packet will be delivered to LAR which is the gateway for the local domain. After receiving the packet, LAR will restore the packet's source address with MN's RCoA. Through the updating and restoring operation, EFLoM can keep the MN's CN unaware of the change of LCoA.

Figure 3.3: Inter-Domain Traffic Illustration

Figure 3.4 illustrates the path for intra-domain traffic of EFLoM. When MN starts the intra-domain traffic connection, the IP swapping agent of MN will update the packet's source with LCoA. Then, the packet will be sent to WAG which is the first hop of the routing. If WAG finds out that MN's peer is residing within the same local domain, WAG will update the packet's destination with the peer's LCoA. Since LCoA is consistent with the peer's current location, the packet will be delivered to the destination node's network directly. If the peer's WAG receives the packet and it finds that the packet is from MN which resides in the same domain, the peer's WAG will restore the packet's source address with MN's RCoA. Finally, when the packet arrives at the peer, the IP swapping agent of the peer will restore

Figure 3.4: Intra-Domain Traffic Illustration

the packet's destination with its RCoA and deliver it to the upper layer application.

To describe the packet forwarding more clearly, the forwarding pseudo codes for MN, WAG and LAR are presented separately. For MN, the forwarding logic is simple, which is presented in Algorithm 1. When MN's IP swapping agent receives a packet, it will first check the packet's destination address *dst* to see if the *dst* is equal to *LCoA*. If yes, the IP swapping agent then replaces the *dst* with its *RCoA* and sends the packet to the applications in the upper layers. If the packet's source address *src* is equal to RCoA, the packet is sent

out to MN's CN. Hence, the IP swapping agent updates *src* with *LCoA* and sends it out.

```

Input: packet received
Output: packet processed
foreach packet k do
    if dst == selfLCoA then
        | UpdateDst(selfRCoA);
    end
    if src == selfRCoA then
        | UpdateSrc(selfLCoA);
    end
end

```

Algorithm 1: MN's Packet Forwarding Algorithm

Algorithm 2 presents the packet forwarding algorithm for WAG. If MN sends a packet to its peer that also resides in the local domain, the packet will arrive at WAG. When the WAG receives a packet, it will first check whether the source node resides in its network via function *NodeinSelfNetwork(src)*. WAG will then check the packet's destination node using function *NodeinDomain(dst)*.

If the source node resides in the WAG's network and the destination node resides in the local domain, the WAG will replace the packet's destination with *peerLCoA*. Since the peer's LCoA is topologically correct and reflects the current location of the peer, the packet will be delivered to MN's peer via the optimal path. However, before the packet arrives at the peer, the packet will be received by the peer's WAG. The peer's WAG will check both source address *src* and destination address *dst*. If the source node resides in the local domain, Function *NodeinDomain(src)* returns true and if the destination node resides in its network, Function *NodeinSelfNetwork(dst)* returns true and the peer's WAG will update the packet's source address with MN's RCoA. For other packets which do not belong to the intra-domain traffic, WAG will just forward them to their destination addresses.

Lastly, the LAR's forwarding algorithm is presented in Algorithm 3. When LAR receives a packet, it will check the packet's source address *src* and destination address *dst*. If the

Input: packet received

Output: packet sent out

```

foreach packet k do
  if NodeinSelfNetwork(src) then
    if NodeinDomain(dst) then
      mn = FindNode(src) ;
      peer = FindNode(dst) ;
      if mn == NULL then
        | FetchNodeFromLAR(src) ;
      end
      if peer == NULL then
        | FetchNodeFromLAR(dst) ;
      end
      UpdateDst(peer LCoA) ;
    end
  end
  if NodeinDomain(src) then
    if NodeinSelfNetwork(dst) then
      mn = FindNode(src) ;
      peer = FindNode(dst) ;
      if mn == NULL then
        | mn = FetchNodeFromLAR(src) ;
      end
      if peer == NULL then
        | peer = FetchNodeFromLAR(dst) ;
      end
      UpdateSrc(mn RCoA) ;
    end
  end
end

```

Algorithm 2: WAG's Packet Forwarding Algorithm

destination node dn resides in the local domain and the source node does not reside in the local domain, the LAR will replace the packet's destination address with dn 's LCoA. If the source node sn resides in the local domain and the destination node does not reside in the local domain, the LAR will update the source address with sn 's RCoA.

```

Input: packet received
Output: packet sent out
foreach packet  $k$  do
    if NodeinDomain( $dst$ ) then
         $dn = \text{FindNode}(dst)$  ;
         $sn = \text{FindNode}(src)$  ;
        if  $sn == NULL$  then
            UpdateDst( $dnLCoA$ ) ;
        end
    end
    if NodeinDomain( $src$ ) then
         $dn = \text{FindNode}(dst)$  ;
         $sn = \text{FindNode}(src)$  ;
        if  $dn == NULL$  then
            UpdateSrc( $snRCoA$ ) ;
        end
    end
end

```

Algorithm 3: LAR's Packet Forwarding Algorithm

After illustrating the data forwarding procedure, there is one issue that needs to be resolved - how WAG knows the RCoA and LCoA when forwarding intra-domain traffic for MN (WAG does not know RCoA and LCoA when forwarding inter-domain traffic). Figure 3.5 presents the signaling between WAG and LAR. When MN starts an intra-domain session with its CN residing in the same domain, the first packet of the session will be sent to LAR and LAR will redirect it to CN. If LAR senses that the two communicating peers reside in the same domain, it sends the binding entry of both CN and MN to all WAG. After receiving the binding entry, WAG can judge that the session between MN and CN is an intra-domain session. Then, WAG performs the forwarding function and delivers packets via the optimal path between MN and CN. To keep the binding entry in WAG synchronized, LAR re-sends

Figure 3.5: WAG Acquires RCoA and LCoA of Intra-Domain Communication Peers

the latest binding entry to all WAG when MN or CN update their binding entry.

3.4 Qualitative Comparative Evaluation

In this section, the three local domain mobility management protocols, namely, HMIPv6, PMIPv6 and EFLoM will be compared. As shown in Table 3.1, our comparison comprises five parts.

First, the entities in the three local mobility management protocols are listed. HMIPv6 has two entities involved, namely, MAP and MN, while PMIPv6 has three network entities involved in the mobility management procedure, which are LMA, MAG and MN. Under

EFLoM, there are also three network entities participating in the mobility related processing. They are LAR, WAR and MN.

Next, the roaming technologies and the traffic overhead introduced by the three mobility management protocols are compared. Both HMIPv6 and PMIPv6 employ IP tunnel technology, while EFLoM employs IP swapping mechanism to deliver traffic in the local domain. Although the roaming technology for HMIPv6 and PMIPv6 is the same, the traffic overhead of the two protocols is different. HMIPv6 establishes the tunnel between MAP and MN, but PMIPv6 only establishes the tunnel between LMA and MAG. Hence, every data packet of HMIPv6 suffers one extra IP tunnel header on both the wired and wireless links while that of PMIPv6 only suffers traffic overhead on wired link. Since EFLoM employs the IP header swapping mechanism to deliver traffic, there is no traffic overhead for data packets of EFLoM.

Next, MN's IP address used in the three mobility management protocols is discussed. Under HMIPv6 and EFLoM, every MN has two IP addresses. One is RCoA and the other is LCoA. RCoA is used by MN as identification and it does not change during MN movement in the local domain. The LCoA is the reachable point for MN and it changes whenever MN moves to a new network. However, under PMIPv6, the LMA and MAG handle mobility signaling on behalf of MN. Hence, MN's IP address will not change in PMIPv6 domain. Although EFLoM uses more IP addresses than PMIPv6, it is not deemed as an issue since IPv6 has adequate IP address resource.

Lastly, the protocol signaling cost is compared. In this chapter, the protocol signaling cost is defined as the packet size of the protocol signaling packets. From Table 3.1, it can be seen that for every handover procedure, HMIPv6 will introduce protocol signaling cost on both the wireless and the wired links. However, for both PMIPv6 and EFLoM, there is no signaling cost in the wireless link and the protocol data is only transmitted in the wired network. Hence, PMIPv6 and EFLoM do not incur wireless bandwidth overhead in the

Table 3.1: Qualitative Comparison of HMIPv6, PMIPv6 and EFLoM

	Local Mobility Protocols		
	HMIPv6	PMIPv6	EFLoM
Participating Entities	MAP and MN	LMA, MAG and MN	LAR, WAG and MN
Roaming Support	IP tunnel technology	IP tunnel technology	IP header swapping mechanism
Traffic Overhead	One IP header on each wireless and wired link	One IP header on wired link and No IP header on wireless link	No IP header on wireless and wired links
IP Change	RCoA does not change, LCoA changes	No IP change	RCoA does not change, LCoA changes
Signaling Cost	Protocol signaling on both wireless and wired links	Protocol signaling only on wired link	Protocol signaling only on wired link

handover operation.

From the comparison, we can conclude that EFLoM has less traffic overhead as well as less wireless signaling cost than the other two peers. Hence, EFLoM can achieve higher performance in terms of handover delay, throughput and signaling cost, all of which are verified in the following experiments.

3.5 Performance Analysis and Simulation

In this section, experiments are conducted using ns2 to compare the performance of HMIPv6, PMIPv6 and EFLoM. As the three mobility protocols are local mobility protocols, the experiments are limited to within a domain. Figure 3.6 shows the experimental scenario. There are three Domain Gateway Routers (DGR1, DGR2 and DGR3) in the network topology and four wireless networks in the local domain. Each network is managed by one Access Router (AR1, AR2, AR3 and AR4). There are ten mobile nodes in each wireless network. For

Figure 3.6: Experiment Scenario

example, MN10 to MN19 are in AR1's network and MN30 to MN39 are in AR3's network. The four networks form a square and each network has two neighboring networks. For example, AR1's network has two neighboring networks, AR2 and AR4. AR2 has two neighbors which are AR1 and AR3.

Before evaluating the performance of EFLoM, an analytical model for MN's movement is developed to facilitate the analysis of the proposal's performance such as signaling cost and handover delay. According to [95], assume every MN's movement follows a random walk model and the probability for MN to cross a domain is p . In a topology where every domain

has c neighboring domains and the total number of domain is N , the Markov transition matrix P is given as follows:

$$P = \begin{pmatrix} 1-p & p/c & p/c & p/c & \cdots & p/c & p/c & 0 & 0 & 0 & \cdots \\ 0 & 1-p & p/c & p/c & \cdots & p/c & p/c & p/c & 0 & 0 & \cdots \\ \cdots & & & \cdots & & \cdots & & & & & \cdots \\ p/c & p/c & p/c & 0 & \cdots & p/c & p/c & p/c & 1-p & \cdots & p/c \\ \cdots & & & \cdots & & \cdots & & & & & \cdots \\ p/c & p/c & p/c & p/c & \cdots & 0 & 0 & 0 & 0 & \cdots & 1-p \end{pmatrix}. \quad (3.1)$$

In each row, there are c nodes with probability of p/c and one node (itself) with a probability of $1-p$.

Then, the probability, Π , for MN to stay in each domain can be deduced as follows:

$$\begin{aligned} & \Pi \times P \\ &= \begin{pmatrix} \Pi_1 & \Pi_2 & \Pi_3 & \cdots & \Pi_{N-2} & \Pi_{N-1} & \Pi_N \end{pmatrix} \times \\ & \begin{pmatrix} 1-p & p/c & p/c & p/c & \cdots & p/c & p/c & 0 & 0 & 0 & \cdots \\ 0 & 1-p & p/c & p/c & \cdots & p/c & p/c & p/c & 0 & 0 & \cdots \\ \cdots & & & \cdots & & \cdots & & & & & \cdots \\ p/c & p/c & p/c & 0 & \cdots & p/c & p/c & p/c & 1-p & \cdots & p/c \\ \cdots & & & \cdots & & \cdots & & & & & \cdots \\ p/c & p/c & p/c & p/c & \cdots & 0 & 0 & 0 & 0 & \cdots & 1-p \end{pmatrix} \\ &= \begin{pmatrix} \Pi_1 & \Pi_2 & \Pi_3 & \cdots & \Pi_{N-2} & \Pi_{N-1} & \Pi_N \end{pmatrix} \\ &= \Pi. \end{aligned} \quad (3.2)$$

Hence, MN's distribution probability for every domain is:

$$\Pi_i = \begin{cases} \Pi_1 * p/c + \cdots \Pi_i * (1 - p) + \Pi_{N-i} * p/c + \cdots \Pi_N * p/c, & 0 < i < c, \\ \Pi_{i-c} * p/c + \Pi_{i-c+1} * p/c + \cdots \Pi_i * (1 - p), & c \leq i < N - c, \\ \Pi_i * (1 - p) + \cdots \Pi_N * p/c + \Pi_1 * p/c + \cdots \Pi_{c+i-N} * p/c, & N - c \leq i < N. \end{cases} \quad (3.3)$$

Then, the probability for MN to stay in each domain is:

$$\Pi = \begin{pmatrix} 1/N & 1/N & 1/N & \cdots & 1/N & 1/N & 1/N \end{pmatrix}. \quad (3.4)$$

3.5.1 Route Hops

The first performance to be evaluated is the traffic route hops. The route hop is the number of routers traversed by the traffic before arriving at the destination. The fewer the route hops, the faster the traffic can be delivered. Hence, the route hop is used as a measure of the transmission delay. One main aim of EFLoM is to provide optimal routing path for intra-domain traffic. This section will present the route hops for intra-domain traffic under the three protocols. At the same time, to compare the transmission delay of the inter-domain traffic, the route hops for inter-domain traffic under the three protocols will be presented.

For inter-domain traffic, the route hop of the three protocols is equal. When a packet is sent out from CN, the DGR of the domain (e.g., DGR1 in the experimental setup) will receive the packet first. Then, the DGR will deliver the packet to the new AR which MN is currently attached to. The AR will then transmit the packet to MN via the wireless link. Hence, the route hop equation for HMIPv6, PMIPv6 and EFLoM is formulated as per Equation (3.5),

$$H^{HMIPv6} = H^{PMIPv6} = H^{EFLoM} = H_{CN-DGR} + H_{DGR-AR} + H_{AR-MN} \quad (3.5)$$

where H_{X-Y} is the route hop from node X to node Y .

For the intra-domain traffic, the hop counts for the three protocols are different. Under HMIPv6 and PMIPv6, the packet needs to be received by DGR. Then, DGR will encapsulate

the packet and deliver it to MN's current location. Hence, the intra-domain traffic will suffer from triangular routing problem under HMIPv6 and PMIPv6. However, for EFLoM, the route hop of intra-domain traffic is different. If a mobile node (MN1) wants to send a packet to another mobile node (MN2) under EFLoM, the packet will arrive at AR1. Then AR1 will transmit the traffic to the AR2 where MN2 resides. Finally, AR2 will deliver the packet to MN2. As the packet is transmitted along the shortest path, the route hop of EFLoM is less than HMIPv6 and PMIPv6. Equations (3.6), (3.7) and (3.8) show the route hop for HMIPv6, PMIPv6 and EFLoM, respectively,

$$H^{HMIPv6} = H_{MN1-AR1} + H_{AR1-DGR1} + H_{DGR1-AR2} + H_{AR2-MN2}, \quad (3.6)$$

$$H^{PMIPv6} = H_{MN1-AR1} + H_{AR1-DGR1} + H_{DGR1-AR2} + H_{AR2-MN2}, \quad (3.7)$$

$$H^{EFLoM} = H_{MN1-AR1} + H_{AR1-AR2} + H_{AR2-MN2}. \quad (3.8)$$

It is to be noted that for PMIPv6, [69] has mentioned that MAG can be configured to bridge mobile nodes when both of them are residing in the same MAG network. The local routing optimization mechanism can reduce the transmission delay. However, if the intra-domain communication resides within different MAG networks, their packets need to be sent to LMA. Then, LMA will deliver the packets to CN's network. In this sense, the average route hop of PMIPv6 is less than HMIPv6.

Experiments are conducted on the three protocols. In the experiment, MN10 moves randomly among the four networks. During its movement, CN0 sends inter-domain UDP traffic to MN10 and MN20 sends intra-domain UDP traffic to MN10. Figure 3.7 shows the average route hops of the three protocols. From Figure 3.7, it can be seen that the route hop of inter-domain traffic is equal for the three protocols. However, the route hop of intra-domain traffic is different. EFLoM has the least route hops and PMIPv6's average route hop is less than HMIPv6's. Figure 3.8 shows the average route hops of each packet received by MN10. The parameter δ is the ratio of intra-domain traffic F_{intra} to the sum of inter-domain

Figure 3.7: Route Hops for Inter- and Intra- Domain Traffic

traffic F_{inter} and intra-domain traffic F_{intra} , which means $\delta = F_{intra} / (F_{inter} + F_{intra})$. From the figure, it can be seen that EFLoM's average route hop is less than the other two protocols.

3.5.2 Protocol Signaling Cost

The second performance metric evaluated is the protocol signaling cost. The signaling cost is the amount of packets which are used by MN to update its location. It includes router solicitation message, router advertisement message, binding update message and binding acknowledgement message. Table 3.2 lists the protocol message size for the three mobility protocols. It also represents the protocol signaling cost incurred by MN for each change of network.

In this section, the term, protocol signaling cost (C_S), is used to describe the additional cost incurred from the handover procedure. Usually, C_S is related to the following four

Figure 3.8: Average Route Hops

parameters.

1. p is the probability that one handover happens in a unit time t . Since the handover probability is p , the staying probability for MN, $\Pi_{MN} = 1 - p$, is as shown in Equation (3.4).
2. s is the total size of the protocol packets in a handover operation.
3. m is the number of mobile nodes in the local domain.
4. t is the unit time.
5. n is the number of handover which happen in a unit time.

With these parameters, the protocol signaling cost C_S can be calculated as per Equation

Table 3.2: Protocol Packet Size

Protocol Name	Packet Type	Packet Size
EFLoM	EFLoM LBU	96 bytes
	EFLoM LBA	96 bytes
	Router Sol	44 bytes
	Router Adv	68 bytes
	Binding Entry Update	142 bytes
HMIP	HMIPv6 RBU	80 bytes
	HMIPv6 RBA	60 bytes
	Router Sol	44 bytes
	Router Adv	68 bytes
PMIP	PMIPv6 PBU	88 bytes
	PMIPv6 PBA	88 bytes
	Router Sol	44 bytes
	Router Adv	68 bytes

(3.9):

$$C_S = \sum_{n=1}^{\infty} n * p^n * (1 - p) * m * \frac{s}{t}. \quad (3.9)$$

Figure 3.9 presents the protocol signaling cost on both the wired and the wireless links. In the simulation, 40 mobile nodes move between the four networks. The MN's movement follows a random walk model. Every 10 seconds, MN chooses to stay in the previous network or move to the neighboring network. The total simulation time is 2000 seconds. From Figure 3.9, it can be seen that for the protocol signaling cost on the wired link, EFLoM is a little higher than PMIPv6 and HMIPv6. This is because the packet size of LBU and LBA is bigger than the signaling packet of the other two protocols (Refer to Table 3.2 for the signalling packet sizes of the different protocols). Moreover, EFLoM needs to send the binding entry update message to WAG along the wired link. The update message is used to support the optimal delivery for intra-domain traffic which is illustrated in previous section. Hence, for wired link, EFLoM cost more than the other two protocols. However, for the wireless link which has much less bandwidth than the wired link, the signaling cost of EFLoM and PMIPv6 is less than HMIPv6. This is because under EFLoM and PMIPv6, MN just employs Router Advertisement and Router Solicitation messages to complete a handover operation.

Figure 3.9: Protocol Signaling Cost

The location updating is done by AR on behalf of MN. Under HMIPv6, MN has to send a location update message to MAP on its own.

3.5.3 Handover

When MN handovers from one network to another, there is a period during which MN cannot receive its traffic. This period is called the *Handover Delay*. Generally speaking, the handover delay D_{HO} is due to three reasons. Firstly, when MN moves from one network to another, its connection with the previous wireless link will be lost. Hence MN will need to search the available wireless link and attach to the new network's wireless link. D_{L2} is used to represent the delay which is a result of the level-2 link switch operation. After attaching to the new network's link, MN will acquire a new IP address to topologically adapt to its new location. D_{IP} is used to represent the time for MN to acquire IP connectivity in the new network. It is the major contribution to the handover delay. However, there are several mechanisms to shorten D_{IP} . For example, the optimized stateless Address configuration [96, 97] allows MN to acquire an IP address without waiting for a Duplicate Address Detection (DAD) period. This can reduce D_{IP} and is the mechanism adopted here for MN to configure the on-link address in the new network. Lastly, MN will send a location update

message to the agent after acquiring the IP address in the new network. Here, D_{LU} is used to represent the time taken to complete the location-update operation. Actually, D_{LU} is mainly determined by the distance between MN and its agent. The local mobility protocols manage to shorten D_{LU} by employing a proxy home agent in its domain. The handover delay is given in Equation (3.10):

$$D_{HO} = D_{L2} + D_{IP} + D_{LU}. \quad (3.10)$$

In this experiment, all the three protocols are deployed in the same network. It means that the D_{L2} of the three protocols are equal. However, the D_{IP} is different for the three protocols. Firstly, the PMIPv6's D_{IP} is equal to zero because PMIPv6 does not change MN's IP address in the new network. Then, EFLoM and HMIPv6's D_{IP} are greater than zero because MN needs to configure a new address under both protocols. When MN moves to a new network, MN needs to wait for the router advertisement message to auto-configure its address. The router advertisement message is sent out by AP every A interval. Hence, D_{IP}^{HMIPv6} and D_{IP}^{EFLoM} are random values and if assuming MN's move-in time is uniformly distributed, the average value of D_{IP}^{HMIPv6} and D_{IP}^{EFLoM} is equal to $\frac{A}{2}$.

The third part of handover delay D_{LU} is also different for the three protocols. HMIPv6 exchanges the location-message between MN and MAP. At the same time, HMIPv6 needs to establish a tunnel before finishing the handover procedure. In this experiment, it is assumed that the tunnel establishing time W_{MN-MAP} is equal to the Round-Trip Time between MN and MAP. Hence, the D_{LU} of HMIPv6 is equal to $2 * T_{MN-MAP} + W_{MN-MAP}$. The T_{MN-MAP} is the transmission time between MN and MAP and the W_{MN-MAP} is the tunnel establishment time between MN and MAP. Similarly, D_{LU} is used for PMIPv6 which exchanges update message between MAG and LMA. Since EFLoM does not employ tunneling between MN and LAR, the D_{LU}^{EFLoM} is equal to $2 * T_{MN-LAR}$.

Hence, the handover delay formula for HMIPv6, PMIPv6 and EFLoM can be shown as

per Equations (3.11), (3.12) and (3.13), respectively,

$$D_{HO}^{HMIPv6} = D_{L2} + \frac{A}{2} + 2 * T_{MN-MAP} + W_{MN-MAP}, \quad (3.11)$$

$$D_{HO}^{PMIPv6} = D_{L2} + 0 + 2 * T_{MAG-LMA} + W_{MAG-LMA}, \quad (3.12)$$

$$D_{HO}^{EFLoM} = D_{L2} + \frac{A}{2} + 2 * T_{MN-LAR}, \quad (3.13)$$

where A is the interval between neighboring advertisement messages, T_{X-Y} is the transmission time between node X and node Y , and W_{X-Y} is the tunnel establishment time between node X and node Y .

To measure the performance of handover delay, UDP flow is used in the experiment. Figure 3.10 shows the performance of UDP flow under the three protocols. When MN hands over from one network to another, there is a period that MN cannot receive the traffic. The longer the handover delay, the more packets are lost in the handover operation. Hence, the receiving rate of MN is used to illustrate the handover delay of the three protocols. From Figure 3.10, it can be seen that EFLoM's delay time is the least and PMIPv6's handover delay is greater than HMIPv6's.

3.5.4 Traffic Overhead

Lastly, the traffic overhead of the three protocols is compared. Both HMIPv6 and PMIPv6 employ IP tunnel technology to forward traffic. The tunnel head will become the overhead of the user data in the network. The traffic overhead $C_{overhead}$ of the three mobility management protocols is defined as $C_{overhead} = L_{IPheader} * H$, where $L_{IPheader}$ is the length of the IP tunnel and H is the transverse hop count for the packet in the local domain.

If the data rate is R bps and the packet size of user data is U , the overhead cost for HMIPv6, PMIPv6 and EFLoM is calculated as per Equations (3.14), (3.15) and (3.16) re-

Figure 3.10: Handover Delay For UDP Flow

spectively,

$$C_{overhead}^{HMIPv6} = L_{IPheader} * H_{MAP-MN} * \frac{R}{U}, \quad (3.14)$$

$$C_{overhead}^{PMIPv6} = L_{IPheader} * H_{LMA-MAG} * \frac{R}{U}, \quad (3.15)$$

$$C_{overhead}^{EFLoM} = 0. \quad (3.16)$$

It is obvious that EFLoM has the least traffic overhead among the three local mobility protocols.

3.6 Conclusion

In this chapter, a new efficient framework for Local Mobility Management (EFLoM) to handle the local mobility management for mobile nodes has been proposed. To avoid introducing traffic overhead into the local domain, EFLoM employs an IP mapping mechanism to deliver MN's traffic. All the packets' destination address is updated with MN's newest IP address by LAR before they enter the local domain. Then, the data packets are restored by WAG with MN's previous IP address and sent to MN. In addition, to evaluate the performance of EFLoM, extensive simulations are conducted to compare with related local mobility management protocols, namely, HMIPv6 and PMIPv6. The compared metrics include route hops, protocol signaling cost, handover delay and traffic overhead. Analysis and simulations show that EFLoM provides a good alternative to manage mobility in a local domain.

Part II

Inter-Domain Handover

Chapter 4

Traffic Distributor Inter-Domain Mobility (TDIDM)

As local mobility can provide users with a better handover support within a domain, the local mobility protocols are gaining more and more popular. However, the local mobility can only manage MN's reachability within a local domain ¹. If MN moves beyond the border of the local domain, the mobility support will be broken. To provide MN continuous support across domains, a solution to interconnect neighboring local domains is proposed. A new network entity called traffic distributor (TD) is introduced. The TD is used to deliver the cross-domain traffic. If MN moves across domain borders, LMA will notify the TD and the TD will redirect MN's traffic to the new domain. Experiments are conducted to compare TDIDM with Neumann's proposal [77, 78] which is another proposal to handle inter-domain handover. Results show that our proposal is a feasible alternative for inter-domain handover,

¹The major difference between intra-domain and inter-domain mobility is the application scenarios. Intra-domain mobility allows users to roam in a limited area (referred to as the local domain) while the latter allows users to roam within a larger range. From the research perspective, a gateway router is usually present to connect the local domain to the Internet as well as to handle functions such as traffic delivery and location tracking. On the other hand, inter-domain mobility needs to address path selection issues due to traversing of the different domains.

and it outperforms Neumann's proposal in terms of binding cache entry number, transmission delay and handover delay.

4.1 Introduction

Next Generation Network is believed to be an all-IP network wireless network (e.g., 3G network [98, 99] and 4G network [100, 101]), which means all existing networks will evolve to a unified network based on IP protocol. During the unification progress, IP mobility management is one crucial issue. With the help of mobility protocols, users can move to any place while maintaining their session undisturbed. Through several years' of development, current mobility system has evolved to a two-level hierarchical system. One is local mobility which provides handover support to MN within a domain. The other is global mobility which allows users to move universally. When users want to manage their reachability within a domain, they can employ local mobility protocols. If they move beyond the border of a domain into another neighboring domain, they have to employ global mobility. However, the global mobility protocols do not perform very well and they suffer issues such as triangular routing [79] and long handover latency.

To enable MN to maintain their sessions while moving across neighboring domains, inter-domain mobility is proposed. Unlike global mobility protocols which employ HA (e.g., MIP) or DNS (HIP) to maintain the latest location of MN, inter-domain mobility employs the previous domain router to handle the movement. As previous domain router has a shorter distance from MN, the inter-domain mobility can provide better handover support to users and integrate neighboring domains together.

In this chapter, a new strategy to enable MN to move across local domains is presented. Through extending the local mobility protocol to neighboring domains, multiple local domains can be connected. To help MN roam across neighboring local domains, some ad-

ditional functionalities in the routers of the aggregate network have been introduced. These routers are responsible for delivering traffic to the domain which MN is attached to. These routers are termed as Traffic Distributors (TD) and the proposal called Traffic Distributor Inter-Domain Mobility (TDIDM). When an inter-domain handover happens, TD will redirect the traffic to the new domain. Then, the serving LMA will tunnel the packets to MAG just as [69]. With the cooperation of TD and LMA, TDIDM can successfully deliver the packets to MN's current location. At the same time, to evaluate TDIDM's performance, it is being compared to Neumann's proposal [78] which also aims to handle the inter-PMIPv6-domain mobility for MN.

The main contributions of TDIDM are as follows. Firstly, it connects the series of neighboring local domains together and allows MN to obtain network-based mobility support within a much larger area. Secondly, it reduces the burden on the domain gateway router. The average number of BCE number in the gateway router is almost 50 % less than that of Neumann's proposal. Thirdly, in TDIDM, the handover delay and transmission delay are shorter than Neumann's. All of these are validated via simulations.

Since Neumann's Proposal is a protocol extended from PMIPv6 protocol, TDIDM is also based on PMIPv6 local domain for ease of illustration and comparison. It is to be noted that although TDIDM is based on PMIPv6, it can be easily extended to other local mobility such as EFLoM and Cellular IP as long as their domain is connected with Internet via a domain gateway router.

The rest of this chapter is organized as follows: Neumann's proposal for the inter-PMIPv6-domain handover is described in Section 4.2 followed by TDIDM and the details of connecting the neighboring PMIPv6 domains together in Section 4.3. In Section 4.4, TDIDM is compared with Neumann's proposal using theoretical analysis and simulation. Section 4.5 concludes this chapter.

4.2 Related Work

Before introducing Neumann's proposal, three new terms: Home Local Mobility Anchor (HLMA), Previous Foreign Local Mobility Anchor (PFLMA) and New Foreign Local Mobility Anchor (NFLMA) are introduced. The three terms are extended from the concept of Local Mobility Anchor (LMA) and are used to differentiate the registering domain, the previous domain and the current domain for MN respectively.

Neumann's proposal is extended from PMIPv6. Under Neumann's proposal, MN first acquires its address configuration in the HLMA domain. Then, MN registers itself with HLMA and starts to be serviced in the HLMA domain. If MN moves from one PMIPv6 domain into another, the previously residing domain is MN's PFLMA domain and the newly visited domain is MN's NFLMA domain. When MN attaches to NFLMA's wireless link, Mobile Access Gateway (MAG) will send a register Proxy Binding Update (PBU) message to NFLMA on behalf of MN. Upon receiving the PBU message, NFLMA will send a PBU to the HLMA to query MN's address configurations. After getting the answer from HLMA, NFLMA will reply with a Proxy Binding Acknowledgement (PBA) to MAG, carrying MN's address configuration. As PBA does not change in the NFLMA domain, MN's address will remain the same as in the HLMA domain. It is noted that if MN returns to its HLMA domain, there is no query operation and HLMA will reply with a PBA to MAG directly.

After the signaling procedure is completed, MN can begin to receive traffic in the NFLMA domain. Under Neumann's proposal, the packet transmission is divided into two steps as shown in Figure 4.1. First, the packet is delivered from HLMA to NFLMA. Then, NFLMA delivers the packet to MN within its domain. According to Neumann's signaling, MN's IP address can be kept unchanged during the movement. Thus, HLMA continuously serves as MN's topological point and HLMA will receive the traffic on behalf of MN. When HLMA receives a packet destined at MN, it will check MN's location and tunnel the packet

Figure 4.1: Signaling Flow For Neumann's Proposal

to NFLMA. After NFLMA unpacks the packet, NFLMA will redirect the packet to MAG whose link is currently attached to by MN and it will forward the packet to MN.

4.3 Traffic Distributor Inter-Domain Mobility

Although Neumann's proposal can provide cross-domain mobility support to MN, there are some problems. First of all, in Neumann's proposal, LMA plays both HLMA role and NFLMA role. Consequently, LMA has to keep BCE for two kinds of MN. The first kind of MN is the one who registers itself in this domain. As MN's HLMA, LMA has to keep the

BCE for MN no matter which domain MN resides. In addition, LMA also needs to keep the BCE for MN who is currently visiting its domain. Hence, under Neumann's proposal, the number of BCE is increased. If there are many MN visiting the domain, the number of BCE will become a burden for LMA and will limit the serving range of LMA. The second issue is the triangular routing problem. The triangular routing problem is similar to the MIP problem analyzed in [79]. In Neumann's proposal, the data packet is transmitted along the path from HLMA to NFLMA, and then to MAG. When MN moves to FLMA's domain which is far from HLMA, the distance between HLMA to FLMA will incur a greater transmission delay for data traffic. In order to overcome these disadvantages, a new alternative solution to address the inter-domain handover challenges is proposed.

A new network entity called traffic distributor (TD) is presented. TD is a router of the aggregate network except that it can handle the cross-domain mobility signaling. As a router, TD is responsible for connecting different PMIPv6 domains together and advertising these domains' network prefixes to the Internet. For example, in the aggregate network shown in Figure 4.2, TD3 and TD4 connect HLMA, PFLMA and NFLMA together. TD1 and TD2 advertise the three domains' network prefixes to the internet.

Other than connecting these domains together, TD needs to handle the cross-domain mobility signaling. When a cross-domain handover occurs, TD in the aggregate network will receive a message from NFLMA. The message is used to inform TD about MN's location. After TD receives the message, it will establish a tunnel with the NFLMA. When TD receives a data packet destined at MN, it will redirect MN's data packets to the NFLMA via the tunnel between them. To make sure MN's traffic can be received by TD, an algorithm to select the appropriate TD set is developed and it will be detailed in the later section. The algorithm is based on current network infrastructure and it does not require any change in the network infrastructure. Hence, TDIDM can be deployed through software-upgrading of the router in the current network. Actually, TDIDM does not require all aggregate routers to be upgraded.

Figure 4.2: The Architecture of TDIDM

It is proposed that only those edge routers which advertise the network prefixes to the Internet (such as TD1 and TD2 in Figure 4.2) and those bifurcating routers which connect PMIPv6 domains together (such as TD3 and TD4 in Figure 4.2) need to be upgraded.

TD signaling procedure is presented in Figure 4.3. Whenever MN enters a new PMIPv6 domain, MAG will first sense MN's attachment to its wireless link and send a PBU message to NFLMA as introduced in PMIPv6 [69]. Before replying to the PBA message, NFLMA will query the address of MN's PFLMA and send a *PBU Forwarding* message to PFLMA. The method on how to obtain the PFLMA's address is illustrated in the following section.

Figure 4.3: Signaling Flow for TDIDM

When PFLMA receives the *PBU_Forwarding* message, it will acknowledge it and send MN's address configuration information to NFLMA. After getting the *PBU_Forwarding_Ack* message, NFLMA will reply with a PBA message to MAG. The address configuration contained in the PBA message is the same as MN in the PFLMA domain. Through the request-reply operation between NFLMA and PFLMA, TD can keep MN's address unchanged, and thus MN is agnostic about the cross-domain movement with respect to layer-3 attachment. At the same time, in order to reduce the packet loss, PFLMA will deliver MN's data packets to NFLMA until the handover is completed.

After sending the PBA to MAG, NFLMA will select a set of TD and send a *PBU_Relay_Request* message to them. The *PBU_Relay_Request* message is used to help TD establish a forwarding entry for MN. When TD receives MN's traffic, it will redirect the

traffic to NFLMA. A tunnel is established between TD and NFLMA after TD replies with the *PBU_Relay_Ack* message to NFLMA. At the same time, for those TD which are selected by PFLMA but not selected by NFLMA, NFLMA will send out a *PBU_Relay_Cancel* message. Upon receiving the *PBU_Relay_Cancel* message, TD will remove the forwarding entry and stop redirecting MN's traffic. With the help of TD, TDIDM can ensure that the data traffic can be delivered to the NFLMA domain where MN currently resides. Note that when MN returns to its HLMA domain, HLMA will not perform TD selection algorithm. Instead, HLMA will send the *PBU_Relay_Cancel* message to all TD which are selected by PFLMA. After TD has removed the forwarding entry for MN, the packets will be delivered to HLMA because HLMA is the topological anchor of MN's network address. After the traffic arrives at NFLMA, NFLMA will send the packet to the right wireless link being attached to by MN. The in-domain transmission is the same as in PMIPv6 [69].

4.3.1 Communicating with MN's PFLMA

When handling the inter-domain handover, TDIDM encounters two issues. The first issue is when NFLMA receives a register PBU for a new incoming MN, how it can communicate with MN's PFLMA. According to [77, 78], two methods, namely, direct location or indirect location, can be employed to help NFLMA resolve this issue. If the direct location method is used, a common database is needed to store MN's current domain information. However, if the indirect location method is used, NFLMA can indirectly reach the PFLMA by sending a query message to MN's address. In this chapter, NFLMA employs the direct location method to communicate with PFLMA. Whenever the common database receives a query message, it will reply with MN's location. At the same time, to keep the database correct, NFLMA will update MN's location after it handovers from PFLMA domain to NFLMA domain.

4.3.2 Qualified-TD-set Selection Algorithm

The second issue is how to select a qualified TD set for the cross-domain MN. A qualified TD set should meet the following two requirements.

Firstly, a qualified TD set can receive MN's traffic. It means that on every path from Internet to MN's HLMA domain, there is at least one TD along the path. For example, in Figure 4.2, there are two paths from Internet to HLMA: $\{TD1-Router1-TD3-HLMA\}$ and $\{TD2-Router2-TD4-HLMA\}$. There are two candidate TD sets: $\{TD1, TD2\}$ and $\{TD3, TD4\}$. Either set can cover all paths from Internet to NFLMA. For example, TD1 stands in the middle of path $\{TD1-Router1-TD3-HLMA\}$ and TD2 stands in the middle of path $\{TD2-Router2-TD4-HLMA\}$. Hence, the set of $\{TD1, TD2\}$ can receive all of MN's data packets. To avoid the TD is absent from the path, TDIDM requires that all HLMA can process TDIDM related protocol signaling which means that all HLMA are candidate TDs.

Secondly, a qualified TD set should redirect traffic to MN's current location as soon as possible. This means the packet's transmission time from the edge router (ER) to NFLMA is the smallest. Here $T_{ER-TD-NFLMA}$ is defined as the transmission time along the path from ER to NFLMA via TD and we can easily get $T_{ER-TD-NFLMA} = T_{ER-TD} + T_{TD-NFLMA}$. To measure the transmission delay resulted from the TD redirection, the delay cost C_D is developed in Equation (4.1)

$$C_D = \frac{T_{TD-NFLMA} - (T_{TD-HLMA} + T_{HLMA-NFLMA})}{T_{ER-HLMA}}. \quad (4.1)$$

The less the C_D , the less transmission delay is produced by TD redirection.

In order to select the qualified TD set, every LMA has three tables - *Advertisement Table*, *Path Table* and *Latency Table*. Tables 4.1, 4.2 and 4.3 show an example of the topology in Figure 4.2. The *Advertisement Table* contains the ER advertising network prefix to the outside Internet. If an ER publishes HLMA's network prefix, it is the topological anchor point for HLMA. HLMA's traffic will reach this ER before they are delivered to MN. The *Path*

Table 4.1: An Example of Advertisement Table

Network Prefix	Advertiser
2.1.0	TD1
2.2.0	TD1
2.3.0	TD1
2.1.0	TD2
2.2.0	TD2
2.3.0	TD2

Table 4.2: An Example of Path Table

Edge Router	Destination LMA	Path
TD1	HLMA	TD1-Router1-TD3-HLMA
TD1	PFLMA	TD1-Router1-TD3-PFLMA
TD1	NFLMA	TD1-Router1-TD3-NFLMA
TD2	HLMA	TD2-Router2-TD4-HLMA
TD2	PFLMA	TD2-Router2-TD4-PFLMA
TD2	NFLMA	TD2-Router2-TD4-NFLMA

Table shows the path information from ER to the destination domain. The *Latency Table* captures the transmission delay between TD and ER, T_{TD-ER} , or between TD and LMA, T_{TD-LMA} . The transmission delay value is consistent with the metric which is used by the routing protocols (e.g., OSPF). The three tables can be easily constructed through protocol signaling update or manual configuration. Actually, the values in the three tables depend on the network topology and they will not change until the aggregate network changes. Since the aggregate network topology usually remains stable, the three tables will not change frequently. In this chapter, assume the three tables are constructed manually as this is the common practice of Internet service providers.

With the three tables, NFLMA can begin to perform the TD-selection operation. Taking the two requirements into consideration, the TD-selection algorithm can be converted to a minimum cost hitting set problem. The vertex set X comprises all candidate TD and each path is the subset T in the hitting set problem. Therefore, the TD-selection problem can be

Table 4.3: An Example of Latency Table

TD	Destination	Latency
TD1	HLMA	300ms
TD1	PFLMA	300ms
TD1	NFLMA	300ms
TD2	HLMA	300ms
TD2	PFLMA	300ms
TD2	NFLMA	300ms
TD3	HLMA	100ms
TD3	PFLMA	100ms
TD3	NFLMA	100ms
TD4	HLMA	100ms
TD4	PFLMA	100ms
TD4	NFLMA	100ms

formulated as Equation (4.2)

$$\begin{aligned}
& \text{minimize} && \sum_{a \in H} w_a x_a, \\
& \text{subject to} && \sum_{a \in T_i} x_a \geq 1, \quad i = 1, 2, \dots, n, \\
& && x_a \in \{0, 1\}, \quad \forall a \in H.
\end{aligned} \tag{4.2}$$

x_a is an indication for node a and w_a is the weight value for node a . If a node is selected, x_a is assigned to 1; else it is equal to 0. The weight value w_a is equal to C_D of TD a which is shown in Equation (4.3) as

$$w_a = C_D(a). \tag{4.3}$$

After deciding the x_a and w_a , we can get the targeted TD set H . The purpose of TD-selection algorithm is to minimize the cost of Equation (4.2) subject to the sum of TD not less than one, which means that every path T_i is covered by at least one node. The formulation in Equation (4.2) can be solved in Algorithm 4. As the hitting set problem is NP-hard which means it cannot be solved in a polynomial time. Considering the TD-selection is related to the handover delay, the time of selecting a qualified TD set should not be too long. Hence,

we present a greedy solution for the TD-selection. In Algorithm 4, we first calculate v_a for each TD a . The v_a is acquired by w_a divided by the number of path hit by a . Then, sort V in ascending order. If the two TDs a and b have the same v which means $v_a == v_b$, then put the TD a which hit the largest number of path before b . Finally, choose v_a from V and remove the hit path from T until $T == \emptyset$. The chosen TD is the selection set H .

```

H =  $\emptyset$ ;
V =  $\emptyset$ ;
foreach  $a \in A$  do
    |  $v_a = \frac{w_a}{|\{a|a \in T_j\}|}$ ;
end
sort  $V$  in ascending order ( if  $v_a == v_b$ , choose the element with largest  $|\{a|a \in T_j\}|$  );
while ( $T \neq \emptyset$ ) do
    | choose the smallest element  $v_a$  whose relevant  $a$  hit  $(T_j, \dots, T_k)$  in  $T$ ;
    | remove  $v_a$  from  $V$ ;
    | remove  $(T_j, \dots, T_k)$  from  $T$ ;
    |  $H \leftarrow a$ ;
end
return  $H$ ;

```

Algorithm 4: Qualified-TD-set Selection Algorithm

4.4 Performance Analysis and Simulation

Before evaluating TDIDM, an analytical model for MN's movement is developed to facilitate analyzing the proposal's performance in terms of BCE number and transmission delay. According to the random walk model shown in Equation 3.1 - Equation 3.4, we can deduce the probability for MN to stay in each domain as:

$$\Pi = \begin{pmatrix} 1/N & 1/N & 1/N & \dots & 1/N & 1/N & 1/N \end{pmatrix}. \quad (4.4)$$

At the same time, we can calculate the probability that MN moves from its HLMA to FLMA (P_{h2f}), the probability that MN moves from the PFLMA to NFLMA (P_{f2f}) and the

probability that MN returns to its HLMA domain (P_{f2h}), respectively, as follows:

$$P_{h2f} = \Pi_i * p = \frac{p}{N}, \quad (4.5)$$

$$P_{f2h} = \Pi_i * \frac{p}{c} * c = \frac{p}{N}, \quad (4.6)$$

$$P_{f2f} = p - P_{h2f} - P_{f2h} = p \left(1 - \frac{2}{N}\right). \quad (4.7)$$

After getting the analytical model for MN's movement, experiments using ns2 are conducted to evaluate TDIDM and Neumann's proposal. The experimental scenario is shown in Figure 4.4. There are three PMIPv6 domains: Domain1, Domain2 and Domain3. Each domain has a different network prefix. In every domain, there are one LMA, one MAG and ten MN. All the three domains connect to TD3 and TD4 with a 500ms-delay link. At the same time, the three domains are connected to one another. Assume that Domain3 is far away from Domain1 while Domain2 is the neighbor of Domain1. Hence, the link delay between Domain1 and Domain3 is set at 200ms while that of the other two links is set at 100ms. CN1 and CN2 connect to the aggregate network with a link delay of 100ms. TD1 and TD2 are also the Edge Router (ER) for the aggregate network and they advertise the three domains' network prefix to CN1 and CN2. The topology of TD in Figure 4.4 is the usual topology of an aggregate network. TD1 connects to TD3 with a link delay of 100ms, and TD2 connects to TD4 with a link delay of 100ms. The link delay between TD1 and TD2 is 200ms, and the link delay between TD3 and TD4 is also 200ms.

4.4.1 Binding Cache Entry (BCE) Number

The number of BCE is one of the factors that limit PMIPv6's serving range and first the influence of inter-domain handover on LMA's BCE number will be investigated. Although both Neumann's proposal and TDIDM can handle the inter-domain handover issues, they have different impacts on LMA's BCE numbers. When a MN moves out of the HLMA's domain, Neumann's proposal requires both HLMA and FLMA to maintain the routability

Figure 4.4: Experimental Scenario

status (e.g., BCE) for MN. The BCE of HLMA can help HLMA redirect the traffic to the right domain and the BCE of FLMA is used to deliver traffic to MN. Hence, in a domain which provides registering service and visiting service to MN, LMA should maintain two kinds of BCE: (i) LMA needs to maintain BCE for MN which is registered in its domain and never delete these BCE even when MN moves out of its domain; (ii) LMA has to set up BCE for the MNs which are currently visiting its domain. However, under TDIDM, LMA does not need to "play two roles", TDIDM just requires LMA to maintain BCE for MN which resides in its domain. When MN moves out of its domain, the relevant BCE will be

deleted. Hence, under TDIDM, LMA's BCE number is equal to the number of MN which are currently active in its domain.

Assume the domain number is N and each domain has m MN registering and all MN follow a random walk model. According to Equation (4.4), MN has the same probability to stay in each domain. Hence, the total number of MN in each domain is m and the number of MN which resides in its HLMA domain is $\frac{m}{N}$. Then, the number of MN visiting this domain is $m - \frac{m}{N}$.

Equation (4.8) shows LMA's BCE number for Neumann's proposal

$$N_{BCE}^{Neumann} = m + \left(m - \frac{m}{N}\right) = 2m - \frac{m}{N}. \quad (4.8)$$

It is equal to the sum of the number of registering MN which is equal to m and the number of visiting MN which is equal to $m - \frac{m}{N}$. Equation (4.9) shows LMA's BCE number for TDIDM

$$N_{BCE}^{TDIDM} = m \quad (4.9)$$

where m is the number of MN residing in the domain.

To validate the model, experiment using ns2 is conducted as follows. There are ten MNs in each domain and all MN handover to another domain every 20 seconds. The MN's movement follows the random walk model. The total simulation time is 200 seconds and Figure 4.5 presents LMA1's BCE number over simulation time. From Figure 4.5, it can be seen that TDIDM's BCE number is 10.6, and it is less than that of Neumann's proposal's which is equal to 16.6.

4.4.2 Transmission Delay

The second performance to evaluate in the experiment is the packet transmission delay. As Neumann's proposal suffers from triangular routing, MN's traffic will be first received by

Figure 4.5: LMA1's BCE Number

HLMA, and then tunneled from HLMA to NFLMA. However, under TDIDM, MN's traffic does not need to traverse HLMA. Instead, with the help of TD, the traffic can be delivered to FLMA directly.

Equation (4.10) shows the transmission delay for Neumann's proposal

$$T_{TransDelay}^{Neumann} = T_{CN-ER} + T_{ER-TD} + T_{TD-HLMA} + T_{HLMA-NFLMA} + T_{NFLMA-MN}. \quad (4.10)$$

It comprises five parts. After the data packet is sent out by CN, it will first arrive at ER; the time from CN to ER router is T_{CN-ER} . Then ER will send the packet to HLMA which is MN's topology anchor point. Because TD stays in the middle way between ER and HLMA, the transmission delay from ER to HLMA can be divided to two parts: the time from ER to TD T_{ER-TD} and the time from TD to HLMA $T_{TD-HLMA}$. Then, HLMA will encapsulate

the packet and send it to NFLMA with a time of $T_{HLMA-NFLMA}$. At last, NFLMA will take a period of $T_{NFLMA-MN}$ to deliver the packet to MN.

Similar to Neumann's proposal, TDIDM's transmission delay comprises four parts as presented in Equation (4.11)

$$T_{TransDelay}^{TDIDM} = T_{CN-ER} + T_{ER-TD} + T_{TD-NFLMA} + T_{NFLMA-MN}. \quad (4.11)$$

First, the packet is sent from CN to ER with a delay of T_{CN-ER} . Then, ER takes a time duration of T_{ER-TD} to send the packet to TD. After TD receives the packet, it will redirect the packet to NFLMA where MN is currently residing. The transmission delay from TD to FLMA is $T_{TD-NFLMA}$. When FLMA receives the packet, it will deliver the packet to MN with a period of $T_{NFLMA-MN}$.

The transmission delay of the two proposals is illustrated in Figure 4.6. From Figure 4.6, it can be seen that in the same network topology, T_{CN-ER} , T_{CN-ER} and $T_{NFLMA-MN}$ are equal in both protocols. According to Qualified-TD-set selection algorithm, the TD has the smallest delay cost. Considering all HLMA's are candidate TDs, the TD in the qualified TD set is expected to have no more delay cost than HLMA (Or else, the HLMA will be selected into the qualified TD set). Hence, $T_{TD-NFLMA}^{TDIDM} \leq (T_{TD-HLMA}^{Neumann} + T_{HLMA-NFLMA}^{Neumann})$ which means the $T_{TransDelay}^{TDIDM} \leq T_{TransDelay}^{Neumann}$.

To further evaluate the transmission delay, a UDP flow is sent from CN1 to MN10 in the experiment. The reason TCP flow is not used is that TCP has burst sending mechanism that may cause packets to suffer queuing delay. The extra queuing delay may affect the measurement of transmission delay. To test transmission delay in the different domains, MN10 is moved from Domain1 to Domain2 at $t=27s$, then MN10 moves to Domain3 at $t=37s$ and MN10 finally returns to Domain1 at $t=47s$. Through MN10's movement, the transmission delay in the three domains can be measured. The simulation result is shown in Figure 4.7. Instead of depicting the delay time of individual packet, the delay of the packets

Figure 4.6: Illustration of Traffic Flow

received by MN10 is averaged at every one second. From Figure 4.7, it can be seen that during the time period from $t=10s$ to $t=27s$, MN10 resides in Domain1. The transmission delay of the two proposals is equal because they have the same routing path. When MN10 moves to Domain2 at $t=27s$, the transmission delay in Neumann's proposal is increased by 100ms which is the delay time between LMA1 and LMA2. However, the transmission delay of TDIDM is not increased because the traffic is distributed to LMA2 by TD3. Note that the delay time of TDIDM at $t=27s$ is a little higher. That is because when MN10 handovers from Domain1 to Domain2, LMA2 keeps receiving MN's traffic from LMA1 until TD3 establish a BCE for MN. When MN10 continues moving to Domain3 at $t=37s$,

Figure 4.7: Transmission Delay

the transmission delay of Neumann's proposal is increased by 100ms again because of the greater distance between Domain1 and Domain3. On the other hand, for TDIDM, the delay remains unchanged because LMA3 still selects TD3 to distribute the cross-domain traffic. Similarly, due to handover operation, the delay of TDIDM at $t=37s$ is higher. Finally, when MN10 returns to Domain1 at $t=47s$, the delay time of the two proposals are equal again.

4.4.3 Handover Delay

The third metric to evaluate is the handover delay. Handover delay is an important parameter which affects MN's performance during the handover period. If the handover delay is too large, some of MN's session will be broken and it cannot continue the session in the new domain. In this experiment, a constant bit rate (CBR) flow to MN10 is sent and the throughput of MN10 monitored during its movement. From MN10's throughput plot, the amount of

traffic lost during the handover can be observed. More lost traffic indicates longer handover delay.

As the handover operation is different, the handover delay of the two proposals is also different. Figure 4.8 illustrates the handover procedure among the different entities. From Figure 4.8, it can be seen that when an inter-domain handover happens, Neumann's proposal requires NFLMA to send a message to HLMA. Then, HLMA replies with a PBA to NFLMA. Hence, Neumann's proposal's handover delay is determined by the distance between NFLMA and HLMA. TDIDM is different. When MN moves to a new domain, NFLMA will send a request message to PFLMA instead of HLMA. Hence, TDIDM's handover delay is determined by the distance between NFLMA and PFLMA.

Equations (4.12) and (4.13) show the handover delay for the two proposals

$$T_{Handover}^{Neumann} = T_{attach} + 2 * T_{MN-NFLMA} + 2 * T_{NFLMA-HLMA} + T_{tunnel}, \quad (4.12)$$

$$T_{Handover}^{TDIDM} = T_{attach} + 2 * T_{MN-NFLMA} + 2 * T_{NFLMA-PFLMA} + T_{tunnel}, \quad (4.13)$$

where T_{attach} is the time taken by MN to attach to a new link; $T_{MN-NFLMA}$ is the period during which MN starts registration and NFLMA receives register PBU from MAG; and T_{tunnel} is the time to establish a tunnel.

Since NFLMA is at a shorter distance from PFLMA than HLMA, the transmission delay between NFLMA and PFLMA, $T_{NFLMA-PFLMA}$, is less than the transmission delay between NFLMA and HLMA, $T_{NFLMA-HLMA}$. Given the network topology is the same, T_{attach} , $T_{MN-NFLMA}$ and T_{tunnel} are equal. Hence, TDIDM's handover delay, $T_{Handover}^{TDIDM}$ is less than Neumann's handover delay $T_{Handover}^{Neumann}$.

To test the effect of the handover delay on the performance of flows, both UDP flow and TCP flow are sent to MN10. The flow's sending rate is $0.5Mbps$. Figure 4.9 shows the handover delay for the UDP flow. In this experiment, MN10 handovers from Domain1 to Domain2 at $t=27s$, and then moves from Domain2 to Domain3 at $t=37s$. When a handover

Figure 4.8: Illustration of the Handover

happens, the MN10's throughput will be reduced. This is because MN10 cannot receive packets during the handover. Hence, the number of lost packets is used to denote the handover delay time. The more the packet loss, the longer is the handover delay.

The handover performance of the two proposals can be observed from Figure 4.9. When MN10 moves from Domain1 to Domain2 at $t=27s$, the handover delay of Neumann's proposal is $0.6s$ and that of TDIDM is $0.5s$. The reason why TD has a shorter delay is that LMA2 can receive traffic from LMA1 and TD3 during the handover period, but LMA2 can only receive traffic from LMA1. Hence, TDIDM's packet loss is less than Neumann's proposal. When MN10 handovers to Domain3 at $t=37s$, the handover delay for Neumann's propos-

Figure 4.9: Handover Delay for UDP flow

al's is $0.77s$ and that of TDIDM is $0.47s$. Under TDIDM, the handover delay is mainly determined by the distance between LMA2 (PFLMA) and LMA3 (NFLMA). However, the handover delay in Neumann's proposal is mainly determined by the distance between LMA1 (HLMA) and LMA3 (NFLMA). Since LMA3 is at a shorter distance from LMA2 than from LMA1, TDIDM's handover delay is less than that of Neumann's proposal at $t=37s$.

The handover delay for TCP flow is next investigated. Unlike UDP flow, TCP guarantees delivery. Packets which are lost during handover will be retransmitted. Hence, the time TCP takes to recover to a stable state is used as the handover delay. Figure 4.10 shows TCP flow's throughput during the simulation. In this experiment, MN10 moves from Domain1 to Domain2 at $t=27s$. After the TCP flow recovers to a stable rate of $0.5Mbps$, MN10 moves to Domain3 at $t=77s$. From Figure 4.10, it can be seen that under TDIDM, the TCP flow can recover to a stable rate in a shorter time than Neumann's proposal. At $t=27s$, TDIDM's delay

Figure 4.10: Handover Delay for TCP flow

is 18s and that of Neumann's proposal is 30s. However, when MN10 moves to Domain3, TDIDM's delay is 25s and that of Neumann's proposal is 50s.

4.4.4 Protocol Signaling Cost

The last performance parameter to evaluate is the protocol signaling cost. In this chapter, the protocol signaling cost is determined as the extra traffic load introduced by the protocol, which means that protocol signaling cost C can be calculated by $n * S * \frac{h}{t}$; where n is the number of protocol signaling packets and S is the size of the packet. Then, the sum of the protocol packet size is multiplied by the hops between the sender and receiver. The protocol signaling cost is then averaged over time t to arrive at cost $C = n * S * \frac{h}{t}$.

To calculate the signaling cost for Neumann's proposal, the probability of handovers has to be identified first. According to Neumann's proposal, FLMA sends out a PBU message to

HLMA when MN attaches to its domain. In Equations (4.5) and (4.7), the probability that MN returns to its HLMA domain is $\frac{p}{N}$. Hence, the probability that MN moves to FLMA is $p - \frac{p}{N}$. Assuming the number of MN is n , the signaling cost for Neumann's Proposal is calculated as per Equation (4.14)

$$C^{Neumann} = n * (p - p_{f2h}) * (S_{PBU}^{Neumann} + S_{PBA}^{Neumann}) * H_{NFLMA-HLMA}. \quad (4.14)$$

Similarly, the signaling cost of TDIDM can be divided to two parts. When MN returns to HLMA, HLMA will send a PBU_Forwarding message to the PFLMA and send a PBU_Relay_Cancel message to TD. Hence the signaling cost for MN returning to HLMA is $(S_{PBU_Forwarding}^{TDIDM} + S_{Forwarding_Ack}^{TDIDM}) + (S_{PBU_Relay_Cancel}^{TDIDM} + S_{PBU_Relay_CancelAck}^{TDIDM}) * H_{HLMA-TD}$. The S_X^{TDIDM} is the size of packet X and H_{X-Y} is the route hop between node X and node Y . The second part of TDIDM's signaling cost happens in NFLMA. When MN attaches to NFLMA domain, NFLMA will need to send three kinds of messages, namely, a PBU_Forwarding message to PFLMA, a PBU_Relay_Request message to the selected TD, and a PBU_Relay_Cancel message to the TD that is selected by PFLMA but is not selected by NFLMA. Equation (4.15) presents the signaling cost for TDIDM

$$\begin{aligned} C^{TDIDM} = & n * (p - p_{f2h}) * (S_{PBU_Forwarding}^{TDIDM} + S_{Forwarding_Ack}^{TDIDM}) \\ & + n * (p - p_{f2h}) * (S_{PBU_Relay_Request}^{TDIDM} + S_{PBU_Relay_Ack}^{TDIDM}) * H_{NFLMA-TD} \\ & + n * (p - p_{f2h}) * (S_{PBU_Relay_Cancel}^{TDIDM} + S_{PBU_Relay_CancelAck}^{TDIDM}) * H_{NFLMA-TD} \\ & + n * p_{f2h} * (S_{PBU_Forwarding}^{TDIDM} + S_{Forwarding_Ack}^{TDIDM}) \\ & + n * p_{f2h} * (S_{PBU_Relay_Cancel}^{TDIDM} + S_{PBU_Relay_CancelAck}^{TDIDM}) * H_{HLMA-TD}. \quad (4.15) \end{aligned}$$

In this simulation, each domain has ten MN and all MN move as per the random walk model. Each MN handovers to another domain every 20 seconds and the total simulation time is 200 seconds. The signaling costs for Neumann's proposal and TDIDM are measured separately. Table 4.4 shows the result. From Table 4.4, we can see that the signaling cost

of Neumann's proposal is 1.15 kbps and that of TDIDM is 5.25 kbps. Although TDIDM's signaling cost is higher than Neumann's proposal, the higher signaling cost is a good tradeoff for TDIDM's better performance. It is because the signaling cost is incurred on the wired link which usually has a large capacity (as high as 10 Gbps) which is very small relative to the capacity of a wired link.

Table 4.4: Protocol Signaling Cost

Protocol	Packet Type	Packet Number	Signaling Cost
Neumann's Proposal	PBU	206	1.15 kbps
	PBA	206	
TDIDM	PBU_Forwarding	288	5.25 kbps
	PBU_ForwardingAck	288	
	PBU_Relay_Request	411	
	PBU_Relay_Ack	411	
	PBU_Relay_Cancel	277	
	PBU_Relay_CancelAck	277	

4.5 Conclusion

In this chapter, an alternative to Neumann's proposal to allow mobile nodes to move between different PMIPv6 domains is proposed. With the help of a traffic distributor (TD), TDIDM combines the neighboring PMIPv6 domains together and enables mobile nodes to continue their existing sessions when they move across multiple PMIPv6 domains. TDIDM's effectiveness against Neumann's proposal in terms of LMA's BCE numbers, transmission delay and handover delay is evaluated. Through experiments, TDIDM outperforms Neumann's proposal in all three aspects and proves to be a good viable alternative to Neumann's proposal.

Part III

Multipath Scheduling in Wireless Network

Chapter 5

Adaptive Load Balancing Algorithm for Multiple Homing Mobile Nodes

In places where mobile users can access multiple wireless networks simultaneously, a multi-path scheduling algorithm can benefit the performance of wireless networks and improve the experience of mobile users. However, existing literature shows that it may not be the case, especially for TCP flows. According to early investigations, there are mainly two reasons that result in bad performance of TCP flows in wireless networks. One is the occurrence of out-of-order packets due to different delays in multiple paths. The other is the packet loss which is resulted from the limited bandwidth of wireless networks. To better exploit multi-path scheduling for TCP flows, this chapter presents a new scheduling algorithm named Adaptive Load Balancing Algorithm (ALBAM) to split traffic across multiple wireless links within the ISP infrastructure. Targeting at solving the two adverse impacts on TCP flows, ALBAM develops two techniques. Firstly, ALBAM takes advantage of the bursty nature of TCP flows and performs scheduling at the flowlet granularity where the packet interval is large enough to compensate for the different path delays. Secondly, ALBAM develops a packet number estimation algorithm (PNEA) to predict the buffer usage in each path. With PNEA, ALBAM

can prevent buffer overflow and schedule the TCP flow to a less congested path before it suffers packet loss. Simulations show that ALBAM can provide better performance to TCP connections than its other counterparts.

5.1 Introduction

With the development of different wireless technologies such as WiMAX, WiFi and UMTS, the future wireless network environment will be the one where different wireless technologies overlap, compete and complement one another. In such an environment, mobile subscribers may expect to improve their experience by accessing multiple wireless technologies simultaneously. Compared with single path transmission, multipath scheduling has many advantages. For example, multipath can ensure successful communication despite link congestion or link failure. Owing to the unstable characteristic of wireless links, mobile subscribers often face disconnections from wireless networks. The simultaneous access to wireless links can provide robustness to subscribers. If one wireless link is disconnected, subscribers can transmit/receive data through another wireless link. Another advantage of multipath scheduling is the load balancing across different paths. Current wireless link performance is always affected by its limited bandwidth. If there is a multipath scheduling scheme to split traffic among the different wireless links and aggregate their bandwidth, they can support more bandwidth-hungry applications in the wireless networks.

However, splitting traffic across multiple wireless links is not an easy task, especially for TCP flows. One of the major issues is the occurrence of out-of-order packets. Owing to delay disparity among multiple paths, the packets may suffer different transmission delays along different paths. If the earlier packet is delivered along a slower path, it may arrive at the destination later than the subsequent packet which is delivered along a faster path. The arrival of packets with a non-monotonically increasing sequence is termed as *out-of-order*

packets. The occurrence of out-of-order packets has an adverse impact on TCP flows which aim to provide sequential connections to users. When TCP destination receives a group of out-of-order packets, it will treat them as an indication of packet loss and reply with several duplicate acknowledgement packets. Subsequently, the TCP source will reduce the rate of transmitting packets even if the network is not congested.

This chapter proposes a new multipath scheduling algorithm named Adaptive Load Balancing Algorithm (ALBAM). In order to reduce out-of-order packets, ALBAM employs three techniques. Firstly, it splits traffic into units of flowlet which is a group of packets. The flowlet scheduling is a technique that takes advantage of the bursty nature of TCP flows and it can split traffic at a larger interval. Moreover, the flowlet splitting can reduce the scheduling frequency, therefore, reducing the occurrence of out-of-order packets. Secondly, ALBAM employs an effective algorithm to prevent the buffer overflow which is the main reason for out-of-order packets in wireless transmission. Thirdly, ALBAM, being an infrastructure based algorithm, improves on the accuracy of path delay measurement by employing the Access Points (AP) to provide statistical delay measurements in each path. Through the three techniques, ALBAM can achieve better performance than related work such as FLARE [89, 90] and OMS [87, 88].

The rest of this chapter is organized as follows: Section 5.2 describes some related work and Section 5.3 gives a detailed introduction of ALBAM which includes the system model of ALBAM, the statistical measurement, the virtual flowlet division, the packet number estimation algorithm and the path selection algorithm. Simulations are conducted to evaluate the performance of ALBAM as well as OMS and FLARE in Section 5.4. Section 5.5 presents the performance of ALBAM for UDP flow. Section 5.6 concludes this chapter.

5.2 Related Work

A lot of research works have been done to solve the issue of multipath scheduling. For example, Opportunistic Multipath Scheduling (OMS) [87, 88] is a multipath scheduling scheme designed to cope with the time-varying characteristic of the wireless channel. Under OMS, each path is modeled as a stochastic process. When data packet arrives at the splitter, OMS chooses the opportunistic low-delay, high-quality path among all candidate paths and distributes the packet via it. At the same time, to avoid too many packets being sent along the same path, OMS introduces a weight parameter to balance the traffic load across multiple paths. With the weight parameter, OMS can ensure the load ratio on each path so as to comply with the desired load in the long run.

OMS is a packet-based scheme which splits traffic at packet granularity. Every time a packet arrives, the OMS splitter performs the scheduling algorithm to choose an optimal path and distribute the packet to the optimal path. The per-packet splitting can accurately satisfy the load balancing across multiple paths. In addition, the flexible allocation of packet-based schemes can adapt to the variation in path quality. For example, when a packet-based splitter finds that one path is congested, the splitter can distribute the subsequent traffic among other available paths. If the path recovers from the congestion, the splitter can reallocate the packets to it. Since the flow's packets are distributed across multiple paths simultaneously, the packet-based splitter can support bandwidth-hungry flow whose throughput exceeds the bandwidth of a single path.

However, the per-packet splitting has many shortcomings. One of the major issues is the out-of-order packet delivery. Figure 5.1 explains the condition to avoid out-of-order packets. In Figure 5.1, the packet P_1 is sent to Path1 and Packet P_2 to Path2. The delay for Path1 and Path2 are D_1 and D_2 respectively. The interval between P_1 and P_2 is T_1 . Assuming the splitter sends P_1 to Path1 at time T_0 , the arriving time for P_1 is $T_0 + D_1$ and the arriving

Figure 5.1: The Condition to Avoid Out-of-Order Packets

time for P_2 is $T_0 + T_1 + D_2$. If $T_0 + D_1 > T_0 + T_1 + D_2$, packet P_1 arrives later than P_2 and out-of-order packet occurs. Hence, the condition to avoid out-of-order is $T_1 \geq D_1 - D_2$ which means that the packet interval should be large enough to compensate for the delay difference between multiple paths.

For multipath scheduling algorithm, the challenge is the balance between maintaining a flexible allocation and reducing the occurrence of out-of-order packets. Flowlet Aware Routing Engine (FLARE) [89, 90] proposes a new scheduling algorithm at the flowlet granularity. The flowlet comes from the burst characteristic of TCP flows. When TCP source transmits traffic, the amount of traffic sent to the receiver is constrained by the size of the congestion window. If the number of sent packets reaches the size of the congestion window, the TCP source will cease transmitting until it receives an acknowledgement packet from the

Figure 5.2: The Packet Interval vs Burst Interval

destination node. Figure 5.2 illustrates the burst characteristic. After TCP source sends out a burst of packets (Burst 1 in Figure 5.2), it stops sending. When ACKs arrive, the source node begins transmitting Packet 7 again. The waiting time is called *burst interval* (e.g., the interval between Packets 6 and 7 in Figure 5.2) and the interval between neighboring packets within a burst is called *packet interval* (e.g., the interval between Packets 7 and 8 in Figure 5.2).

Since the burst interval is the acknowledgement waiting time which is related to the transmission delay between TCP source and destination, the burst interval is usually much larger than the packet interval. FLARE takes advantage of the burst interval to perform scheduling at the beginning of each burst. The large burst interval provides a good opportunity to compensate for the delay disparity between the different paths. Hence, FLARE can reduce the possibility of out-of-order packets when switching delivery path. Specifically speaking, when a TCP burst arrives, the FLARE splitter chooses the path whose load ratio is furthest away from its desired share and distributes the burst to the path. For the packets within the burst, FLARE does not perform path switching and deliver them along the same

path. Through splitting flow at the granularity of burst, FLARE can reduce the occurrence of out-of-order packets as well as achieve good load balancing across multiple paths. FLARE introduces a different terminology to describe the bursty characteristic of TCP flow. The group of packets sent together is called *flowlet* instead of *burst* and the term *burst interval* is referred to as *flowlet interval*. Hereinafter, the terms, *burst* and *flowlet* will be used interchangeably unless otherwise specified.

Although FLARE provides a good scheduling algorithm to balance the allocation accuracy and out-of-order packets, it has its own disadvantages. First of all, the flowlet interval does not exist all the time. According to our investigation, flowlet interval only exists in the period when the flow rate is unstable. Figure 5.3 presents the results of the simulation carried out to monitor the change of packet interval during the lifetime of TCP flow. The simulation uses two kinds of TCP flows as examples, one flow is NewReno [102–106] and the other one is Tahoe [105, 106]. In the simulation, the transmission delay between the source and the destination is set at 100ms with the flowrate at 1000 kbps and the packet size is 1000 bytes. To make the results clearer, only data points whose values are larger than 30ms are presented, which means only flowlet intervals are reflected in Figure 5.3(a). Packet interval between neighboring flowlet interval which is very small is thus not reflected. From Figure 5.3, it can be seen that flowlet interval exists at two stages. One is from 0 to 10 seconds where TCP flow increases its rate from 0 to 1000 kbps. The other one is between 50th and 60th seconds where packet loss occurs during the transmission. The reason for the disappearance of the flowlet is that the TCP sender does not need to wait for the ACK packets in the steady state. When TCP stays at the slow-start stage (flows begin transmitting or the lost packets cause flows to be retransmitted), the congestion window size is very small and TCP source needs to stop and wait for acknowledgements after sending a burst of packets into path. As the flow continues, the congestion window is increased (at the slow-start stage, TCP increases the size by one segment for each acknowledgement received. At congestion-

Figure 5.3: The Flowlet Interval During The Lifetime of TCP Flow

avoidance stage, TCP increases the size by one segment for every maximum segment size). When the size of the congestion window is large enough, the TCP source will not continue the send-and-wait behavior and the flowlet interval disappears. The second disadvantage for FLARE is that FLARE does not consider the packet loss during the scheduling. As TCP flow is a guaranteed delivery service, it retransmits the packet when packet loss occurs. The retransmission cost is high in FLARE because the path congestion can cause the loss of a burst of packets.

In order to address the two disadvantages of FLARE, ALBAM develops two algorithms- Virtual Flowlet Division (VFD) and Packet Number Estimation Algorithm (PNEA). VFD is a traffic division algorithm. It is used to group several adjacent packets together when flowlets are missing. As Figure 5.3 shows that, when the size of congestion window is large enough, the TCP flow becomes stable and flowlets disappear. With VFD, ALBAM can divide traffic into several groups and schedule these groups across multiple paths. PNEA is a technique used to estimate the number of on-delivery packets in each path. In TCP transmission over wireless networks, packet loss is quite common. To prevent packet retransmission due to packet loss and improve the performance, ALBAM employs PNEA to monitor the path status and avoid sending packets to congested paths.

5.3 Adaptive Load Balancing Algorithm (ALBAM)

5.3.1 System Model and Overview

Adaptive Load Balancing Algorithm (ALBAM) is a multipath scheduling algorithm developed to split TCP across different wireless paths. Figure 5.4 presents a general application scenario for ALBAM. A multiple interface (MIF) node gets access to the network via several wireless technologies (such as WiFi and WiMAX). Both WiFi and WiMAX interfaces can receive traffic simultaneously. The source node of TCP flow resides in the network side and there is a bifurcating node midway between the source and the destination. ALBAM is deployed in the bifurcating node and schedules the traffic to the MIF node across multiple paths. Figure 5.4 actually represents a general model of a local domain which is deemed as the dominant structure for future wireless networks [63, 65]. Since the Domain Gateway Router (DGR) is the topological anchor point of the local domain, it can play the role of a bifurcating node in the model shown in Figure 5.4.

Figure 5.4: The System Model of ALBAM

The operation of ALBAM is divided into 3 parts. Firstly, ALBAM performs delay measurement for each path. The path delay is used by ALBAM to split traffic across multiple paths. When ALBAM decides to switch traffic from one path to another, it calculates the delay difference between two paths and buffers the traffic until the interval is larger than the difference. This operation can minimize if not eradicate out-of-order packets. The second part of ALBAM is the traffic division. ALBAM is a flowlet-based scheduling algorithm. It switches the flow to different paths at the granularity of flowlets. In order to manage the “missing” flowlets shown in Figure 5.3, ALBAM develops an algorithm to divide the flow into several virtual flowlets (VF) when flow enters stable stage. The third part of ALBAM is packet number estimation algorithm (PNEA). In order to prevent buffer overflow during transmission, ALBAM employs PNEA to monitor the buffer usage of the delivery path. If PNEA predicts a buffer overflow will occur, ALBAM switches the traffic to another path immediately.

With the aforementioned operations, ALBAM aims to improve the scheduling effect and aggregate multipath bandwidth more effectively. In the following sections, detailed introduction of the three mechanisms will be given.

5.3.2 Delay Measurement

Before scheduling flows across multiple paths, ALBAM needs to perform delay measurement. Existing scheduling schemes such as FLARE and OMS usually employ a probing method which sends a ping message to each path. The probing method, however, has several disadvantages. First of all, the probing method does not consider the asymmetric issue. In access network, the traffic load of downlink and uplink are usually unequal. When probing method estimates the delay by halving the interval between sending and receiving the ping message, it introduces an error into the delay measurement. Secondly, the probing method may measure a different path delay in a multiple access scenario. For MIF node, the receiving and replying interfaces may not be the same. Actually, to make full use of network resources, MIF node usually employs an intelligent algorithm (such as [107, 108]) to select a better interface and send packets via it. For example, when a MIF node receives a ping message at the WiFi interface, it may reply the ping message via its WiMAX interface because the WiMAX network has better performance. Thus, the different sending and replying paths will introduce a skew to the probing measurement as well.

To improve accuracy of delay measurement, ALBAM suggests a statistical measurement. Under statistical measurement, the path delay comprises three parts. They are delay from splitter to the Access Point (AP), the queuing delay in AP and the propagation delay from AP to the mobile node. ALBAM measures the first part by sending a ping packet between the splitter and the AP. Since both splitter and AP are ISP entities, they can be synchronized and the one-way delay can be easily obtained. Even if the synchronization cannot be implemented, the high bandwidth of wired link between the splitter and the AP can reduce the asymmetric affect and yield a reasonably accurate measurement.

Except the first part of delay measurement, the other two parts of the path delay are measured by the AP. The queuing delay is easily measured by averaging the time the packets stay

in the AP's buffer. The propagation delay is a little complicated, and it differs in different types of networks. If the wireless network is one-hop network, AP can acquire the propagation delay with the support of MAC protocols. For example, in IEEE 802.11 MAC with RTS/CTS, AP can use half of the interval between RTS and ACK to represent the propagation delay. In IEEE 802.16 network, the time allocated for the downlink burst to represent the propagation delay can be used. If the wireless network is a multi-hop network, AP can send a ping message to the mobile node periodically.

To evaluate the performance of probing method and statistical method, experiments are conducted to measure the path delay under the different flow rates. The measurement error E_D is defined as the normalized difference between the measured delay $\tilde{D}$ and the actual transmission delay D , which is shown in Equation (5.1).

$$E_D = \frac{|\tilde{D} - D|}{D}. \quad (5.1)$$

Figure 5.5 shows the errors for the different measurement methods. In the experiment, the path delay suffered by every packet is averaged and the value is used as the actual path delay. Then, different methods are used to measure the delay. From Figure 5.5, it can be seen that the statistical method yields the lowest error in the one-hop network. Even in multi-hop network, the statistical method can yield a better performance than the probing method.

5.3.3 Virtual Flowlet Division

After obtaining the path delay, ALBAM begins to split traffic across multiple paths. As ALBAM manages the traffic at the granularity of flowlets, it needs a traffic division algorithm when the flowlets are missing. Virtual Flowlet Division (VFD) is such an algorithm to help ALBAM divide a series of packets into different groups. These groups are called Virtual Flowlet (VF) and they are basic units for ALBAM to schedule.

Figure 5.5: Delay Measurement

Figure 5.6 presents the operation of VFD. In Figure 5.6, the packets are divided into 3 groups: VF1, VF2 and VF3. The size of each VF is the same (in Figure 5.6, each VF contains 7 packets). Since VFD is applied during the time when a flow becomes stable, the interval between packets is small. If ALBAM schedules these VF to different paths, the interval between VF is not large enough to compensate for the delay difference between the two paths. To reduce the out-of-order packets, VFD buffers the packets until the interval between neighboring VF is larger than the delay difference. For example, if ALBAM wants to send VF1 via Path1 and VF2 via Path2. The two path delays are D_1 and D_2 respectively, ALBAM first calculates the safe threshold (T_S) with $T_S = D_1 - D_2$. Then, ALBAM sets

Figure 5.6: Illustration on Virtual Flowlet Division

the buffering time (T_B) as value $T_B = T_S - I_P$ where I_P is the packet interval between VF1 and VF2. In Figure 5.6, the time gap between Packets 6 and 7 is I_P . After buffering, the interval between Packets 6 and 7 is enlarged. Hence, when ALBAM switches traffic to path2, the possibility of out-of-order packets is reduced. It is to be noted that there is an exception that ALBAM will end the buffering before T_B . If ALBAM finds that the sum of buffered packets and unacknowledged packets exceeds the buffer size in AP, ALBAM will stop buffering and send out the packets in advance. The logic behind this exception is that the cost of retransmission is much higher than the cost of out-of-order packets.

With the help of the buffering operation, ALBAM ensures the virtual flowlet interval satisfy the safe switching threshold. Hence, ALBAM can reduce the out-of-order packets

when switching the delivery path for VF. At the same time, to keep the splitting granularity unchanged, ALBAM sets the frequency of VFD to be equal to the frequency of scheduling flowlet in the unstable stage. With VFD, ALBAM can always trigger the scheduling operation and aggregate the bandwidth of multiple paths. Therefore, ALBAM can improve beyond the performance of FLARE and provide larger bandwidth to applications than FLARE.

5.3.4 Packet Number Estimation Algorithm

VFD helps ALBAM always schedule traffic at the granularity of flowlet. The flowlet has a large interval to compensate for the delay difference between multiple paths. Hence, ALBAM can reduce the out-of-order packets. However, it is not the whole story. Another issue that ALBAM wants to handle is the packet loss. According to investigations done in [109–111], buffer overflow is the major cause of packet loss (the loss due to wireless propagation error can be reduced greatly with a local retransmission mechanism and perhaps forward error correction). To reduce the buffer overflow during transmission, ALBAM develops an algorithm named Packet Number Estimation Algorithm (PNEA). PNEA monitors the buffer usage in each path and alerts ALBAM when there is a possibility of buffer overflow.

The basic idea of PNEA is based on the acknowledgement mechanism of TCP flow. When ALBAM sends out a packet or buffers a packet with VFD, PNEA keeps its sequence number in the delivery path. When ALBAM receives an acknowledgement (ACK) packet from the receiver, PNEA clears the corresponding sequence number from the delivery path. Figure 5.7 presents an illustration of the PNEA operation. Under PNEA, each path maintains a sequence queue. When a packet is scheduled to one path, PNEA extracts the sequence number from the packet's TCP header and inserts it to the queue of the delivery path. For example, in Figure 5.7, packet 19 is scheduled to be delivered via Path1. Hence, the sequence

Figure 5.7: Packet Number Estimation Algorithm

number of 19 is inserted into Path1's queue.

The packet's sequence number is kept in the queue until ALBAM receives acknowledgement. Upon the arrival of ACK, PNEA clears the sequence queue. The way to clear the sequence queue depends on the type of TCP acknowledgement. If the acknowledgement is selective, the clearing operation is simple and PNEA just removes the corresponding sequence number from the queue. If the acknowledgement is accumulative, PNEA removes all sequence numbers which are less than the acknowledgement number. For example, in Figure 5.7, when PNEA receives the ACK for packet 11, the selective TCP removes sequence number 11 from Path2's sequence number queue while accumulative TCP removes sequence numbers 5, 6, 7 in Path1's queue and sequence numbers 8, 9, 10, 11 in Path2's queue. As the majority of TCP flows use accumulative acknowledgement, PNEA focuses on the accumulative one.

Through collecting the unacknowledged packets in the sequence queue, PNEA can estimate the number of on-delivery packets. This works well except when a packet is lost.

Actually, in TCP transmissions, there are two reasons leading to the unacknowledgement. Firstly, the packet is on its delivery path and it has not arrived at the destination yet. Secondly, the packet gets lost in the path. For ALBAM, the former is the concern and the lost packets should be reduced as much as possible. When TCP flow encounters a packet loss, the receiver will reply a duplicate ACK to notify the sender to retransmit. With the duplicate ACK, PNEA can easily differentiate whether unacknowledgement is resulted from on-delivery or packet loss. To better explain the problem, an example is presented in Figure 5.7 where Packet 13 is lost. PNEA receives duplicate ACK12 and it cannot clear sequence numbers 13, 14 and 15 from Path2's queue. Hence, if PNEA makes the estimation based on the unacknowledged packets in the queue, the estimation is 3. However, the actual queue size is 0 because the duplicate ACKs are triggered when Packets 14 and 15 are received. Packet 13 is lost during the transmission.

In order to make the correct estimation, PNEA adds a flag to differentiate the flow statuses, namely, recovery status and non-recovery status. The recovery status refers to a flow which suffers a packet loss and has not recovered from the loss yet. The non-recovery status refers to a flow without packet loss. Corresponding to the flow's two statuses, PNEA uses two methods to estimate the queue size. When flow is in non-recovery status, PNEA estimates the on-delivery packets by counting the queue size. When PNEA receives more than 3 duplicate ACK, PNEA marks the flow as recovery status and stores the queue size into the parameter *Cursize*. Algorithm 5 shows how PNEA estimates the path's buffer. When the flow is in recovery status, PNEA returns *Cursize*. Otherwise, PNEA counts the unacknowledged packets in the queue.

Besides Algorithm 5 which makes estimation based on *Cursize* and sequence queue, PNEA also needs another two algorithms to maintain both *Cursize* and sequence queue when sending out packets and receiving ACKs. Algorithm 6 shows the procedure of inserting a packet's sequence number into the queue. Every time ALBAM distributes a packet to a

```

if flow  $K$  is in recovery status then
  | return  $Cursize$  for Path  $i$ 
end
else
  | count = 0;
  foreach unacknowledged packet in path  $i$ 's sequence queue do
    | increase count by 1;
  end
  | return count;
end

```

Algorithm 5: Estimate the Number of Packets Delivered along a Specific Path i

path, PNEA extracts the sequence number from the packet's TCP header and inserts into the sequence queue. Then, PNEA checks whether the flow is in recovery status (The recovery status is set when PNEA receives more than three duplicate acknowledgement packets and is cleared after the flow retransmits the lost packets successfully). If yes, PNEA increases the path's $Cursize$ by one. Else, PNEA does not update $Cursize$.

```

if TCP packet of flow  $K$  is scheduled to path  $i$  then
  | if flow  $K$ 's HighestSentSeqNo < SeqNo then
    | HighestSentSeqNo = SeqNo;
  end
  | insert the SeqNo into path  $i$ 's queue;
  | if flow  $K$  is in recovery status then
    | the path  $i$ 's  $Cursize$  += 1;
  end
end

```

Algorithm 6: Insert Packet Sequence Number into Queue After Scheduling

Algorithm 7 clears the sequence queue upon the arrival of ACK. When an acknowledgement packet arrives, PNEA compares it with the last acknowledgement packet. If the sequence number ($AckNo$) is larger than the last packet, PNEA clears each path's sequence queue by removing the sequence numbers which are less than $AckNo$. At the same time, PNEA checks the flow's status. If the flow is in the recovery status and the received $AckNo$ is

greater than *RecoverySeqNo*, it means that all the lost packets have been retransmitted successfully. Hence, PNEA clears the recovery status for the flow. If the flow is in the recovery status but *AckNo* is less than *RecoverySeqNo*, it means that only a part of the lost packets have been retransmitted. Hence, PNEA does not clear the recovery status but increases the base of the acknowledgement, *AckBase*, which is used to estimate the sequence numbers of the acknowledgement packets during the recovery stage.

```

if AckNo > LastAckNo then
    clear the sequence queue for each path;
    if flow K is in recovery status then
        if AckNo ≥ RecoverySeqNo then
            | clear flow K's recovery status;
        end
        else
            | AckBase = AckBase + (AckNo - LastAckNo);
        end
    end
end
else if AckNo ≡ LastAckNo then
    DupNo ++;
    if DupNo ≡ 3 then
        mark the flow K's status as recovery;
        Cursize = the size of (sequence queue);
        foreach packet i whose Seq No is between AckNo+2 and AckNo + 3 do
            | decrease the Cursize for the delivery path of packet i;
        end
    end
    if flow K is in recovery status and DupNo ≠ 3 then
        | decrease Cursize for delivery path of packet of
        | (AckNo + DupNo - AckBase);
    end
end
LastAckNo = AckNo;

```

Algorithm 7: Clear Sequence Queue upon Arrival of TCP Acknowledgement

If *AckNo* is equal to the last acknowledgement packet and the number of duplicate packets is equal to three, PNEA sets the flow's status to recovery status and initializes the parameters-*Cursize*. As *Cursize* represents the number of on-delivery packets, PNEA sets *Cursize* with

the size of sequence queue. Moreover, PNEA finds out the delivery path for packets between $AckNo + 2$ and $AckNo + 3$ and reduces *Cursize* of the delivery path. The reason why PNEA ignores the acknowledgement packet $AckNo + 1$ is that PNEA cannot judge whether the packet of $AckNo + 1$ is lost. Another reason which results in three duplicate acknowledgements is due to packets arriving out-of-order at the receiver. Hence, PNEA makes a conservative estimation of the number of packets in each path. Finally, for every duplicate acknowledgement packet, PNEA estimates its sequence number using the formula, $AckNo + DupNo - AckBase$ and reduces *Cursize* of the sequence queue of its delivery path.

With the help of PNEA, ALBAM can estimate the number of on-delivery packets for each path. Actually, PNEA affects ALBAM's scheduling by marking the path with a busy flag. When the number of on-delivery packets reaches the path's *overflow threshold* which is designated by users, PNEA sets a busy flag to the path. It means that a buffer overflow may happen if ALBAM continues sending packets via the path. Hence, ALBAM switches the traffic to another less-congested path immediately. If the number of on-delivery packets falls below *safe threshold*, PNEA clears the busy flag. It means that ALBAM can choose the path as a safe delivery path.

The interval between setting a busy flag and clearing the busy flag is termed as busy period. During busy period, the path may suffer a buffer overflow. Hence, ALBAM should avoid distributing traffic to busy path. However, the busy period is resulted from PNEA's prediction and it may be different from the actual occurrence of buffer overflow. In the following section, prediction period and busy period will be used interchangeably. To illustrate the relationship between PNEA's prediction and the occurrence of buffer overflow, an illustration is presented in Figure 5.8. In Figure 5.8, the T_o is the time that buffer overflow occurs, T_b is the time that PNEA sets the busy flag (also the beginning time of the prediction period) and the T_e is the time that PNEA clears the busy flag (also the end time of the prediction time).

The three kinds of relationship are as follows:

1. The buffer overflow happens within the prediction period ($T_b \leq T_o \leq T_e$ in Figure 5.8.(a) and (b)). In this case, the PNEA makes a correct prediction. However, there is difference between the two subfigures. In Figure 5.8.(a), the T_o is shortly behind T_b which means PNEA makes a prediction with a high accuracy. In Figure 5.8.(b), T_o is far behind T_b . Hence, the prediction accuracy in Figure 5.8.(b) is low and it will result in a low usage of aggregated bandwidth.
2. PNEA predicts a buffer overflow, but there is no occurrence at all(There is no T_o between T_b and T_e in Figure 5.8(c)). In this case, the prediction is considered ineffective and it will result in bandwidth wastage.
3. The buffer overflow happens before the prediction period ($T_o < T_b$ in Figure 5.8(d)). In this case, PNEA makes a wrong prediction.

With the three kinds of relationship between prediction and overflow occurrence, three performance indices, namely, Prediction Correctness γ_C , Prediction Efficiency γ_E and Prediction Accuracy γ_A are defined as per Equations (5.2), (5.3) and (5.4), respectively,

$$\gamma_C = \frac{N}{Z} \quad (5.2)$$

$$\gamma_E = \frac{K}{M} \quad (5.3)$$

$$\gamma_A = \sum_{i=1}^N \frac{(T_o - T_b)}{N} \quad (5.4)$$

where N is the number of packet loss predicted by PNEA correctly, Z is the number of packet loss occurring during the transmission, K is the number of prediction which has at least one overflow happening in its time and M is the number of predictions made by PNEA.

Prediction Correctness (γ_C) is the percentage of buffer overflow which is predicted correctly by PNEA. It is an evaluation related to the occurrence of buffer overflow. The more

Figure 5.8: Buffer Overflow Prediction

buffer overflow falls in the prediction period, the higher γ_C is. Prediction Efficiency γ_E is a parameter which measures the number of empty predictions. It is an evaluation of prediction period. If PNEA sets a busy flag to a path but there is no occurrence of buffer overflow, the prediction is empty (i.e., Figure 5.8(c)) and the γ_E is low. Prediction Accuracy γ_A is a parameter related to the position of buffer overflow. If PNEA makes an early prediction (i.e., Figure 5.8(b)), ALBAM will stop sending packets to the path at an early time and the path's bandwidth resource will be wasted. Hence, the lower γ_A is, the higher bandwidth efficiency ALBAM can achieve.

Among the three performance indices, Prediction Correctness (γ_C) is the most important because it is related to detecting the potential buffer overflow. If PNEA achieves 100% for

Figure 5.9: Prediction Correctness

γ_C , it means that all the buffer overflows are predicted. Figure 5.9 presents the performance of γ_C in the simulations. In the simulations, the path's buffer size is set to 10 packets and the *overflow threshold* is varied. Figure 5.9 presents the performance of γ_C under each overflow threshold. From Figure 5.9, it can be seen that as the threshold increases, γ_C is lowered. An *overflow threshold* of 14 can ensure all packet loss is effectively predicted by PNEA.

The second important performance index is Prediction Efficiency (γ_E). As a parameter related to successful predictions, the higher γ_E is, the fewer empty prediction PNEA makes.

Figure 5.10: Prediction Efficiency

Figure 5.10 presents the relationship between γ_E and *overflow threshold*. From Figure 5.10, it can be seen that under the condition of γ_C being equal to 100%, a higher threshold can help PNEA achieve a higher γ_E . Hence, if unnecessary path switching by ALBAM is reduced, the overflow threshold can be set to a high value (On the condition that γ_C is 100%).

The last performance index is Prediction Accuracy (γ_A) which indicates the relative position between prediction period and the occurrence of buffer overflow. If the buffer overflow happens long after the prediction, PNEA will waste bandwidth resource because ALBAM

Figure 5.11: Prediction Accuracy

will leave the path idle even if the path is not congested. Figure 5.11 presents the relationship between γ_A and occurrence of overflow. From Figure 5.11, it can be seen that the higher the overflow threshold, the lower the accuracy PNEA can achieve. Actually, the value of *overflow threshold* should be set to the highest value on the condition that all packet loss can be predicted (γ_C equals 100%).

After presenting the three performance indices of PNEA, the last concern about PNEA is the cost of the prediction - how much buffer space should be reserved for PNEA. To

Figure 5.12: Buffer Size for PNEA

investigate the cost of PNEA, simulations are conducted and the results are shown in Figure 5.12. The cost of PNEA is mainly related to three elements, namely, the flow rate, the delay from the splitter to the AP and the number of delivery paths. As the number of interfaces in a mobile node is limited, the number of delivery paths is also limited (usually no more than three). Figure 5.12 presents the buffer size with varying flowrate and delay. In the simulation, two delivery paths are defined for the mobile node. From Figure 5.12, it can be seen that the buffer size of PNEA is less than 3.5 kbps.

5.3.5 Path Selection Algorithm

The operation of ALBAM is summarized in this section. Algorithm 7 shows how ALBAM performs the path selection algorithm for a flow. When a packet arrives at the splitter, AL-

BAM judges whether the packet interval falls into the confidence range of the flowlet interval. If yes, ALBAM treats it as the first packet of the flowlet and switches the delivery path for the newly arrived flowlet. The delivery path is the one whose load ratio is furthest away from its desired share. This can balance the load between the different paths. If the packet interval is not equal to the flowlet interval, ALBAM needs to check the flow's status. If the flow is in a stable state (the packet interval varies little), ALBAM divides the traffic into several VF at the frequency as stated in Section 5.3.3. If the flow is not in the stable state, ALBAM treats the packet as the one in the middle of the flowlet. Hence, ALBAM does not switch the delivery path for the packet and sends it along its previous path.

Lastly, ALBAM checks the buffer usage of the delivery path with PNEA as discussed in Section 5.3.4. If the delivery path is under the busy period which means that there is a possibility that packet loss will occur, ALBAM chooses another path which is not in the busy period. If all the candidate paths are in the busy period, ALBAM chooses the path whose load ratio is the furthest away from the desired share. ALBAM then updates the previous path and the load ratio on the delivery path.

5.4 Simulation Results

To evaluate the performance of ALBAM, experiments are conducted using ns2. OMS and FLARE are chosen for comparison as they are all multipath scheduling algorithms. OMS uses a packet-based algorithm while FLARE uses a flowlet-based one. Performance comparison is made against four metrics, namely, *max flow*, *out-of-order packets*, *splitting error* and *transmission time*. *Max flow* refers to the highest throughput that can be supported under the different scheduling algorithms. It reflects the effectiveness of aggregating the bandwidth of multiple paths. *Out-of-order packets* counts the number of packets which arrives

```

if packet interval falls into the confidence range of the flowlet interval then
    | deliverypath = path i which is furthest away from its desired share;
end
else
    | deliverypath = PreviousPath;
    | if flow enters the stable procedure then
    | | if NOW - LastScheduleTime  $\geq$  FlowletInterval then
    | | | deliverypath = path i which is furthest away from its desired share;
    | | | employ the buffering technology to reduce the out-of-order packets;
    | | end
    | end
end
check the status of path i;
if path i is in busy status then
    | find a unbusy path j which is furthest away from its desired share;
    | if all paths are busy then
    | | choose path K which is furthest away from its desired share;
    | end
end

```

Algorithm 8: Path Selection Algorithm

at the destination in the wrong order. The fewer the out-of-order packets, the better is the scheduling algorithm.

Splitting error evaluates the performance of load balancing. An ideal multipath scheduling algorithm should maintain the load on each path as close to the desired share as possible at all times which is practically not feasible. The definition of *Splitting Error* (E_S) is shown in Equation (5.5)

$$E_S = \frac{1}{N} \sum_{i=1}^N \frac{|F'_i - F_i|}{F_i}. \quad (5.5)$$

E_S measures the deviation of the actual load (F'_i) from the desired loading (F_i) in each path and sums up the deviation. The lower E_S is, the better load-balancing the scheduling algorithm can achieve. *Transmission time* measures the performance of the scheduling algorithms on short-time TCP flow. In mobile scenarios which have unstable connection, data is usually transferred in the form of short-time TCP flow. For users on the move, the data transmission time is of more concern than the throughput. Hence, for short-time flow, the

less transmission time it takes, the higher is the performance of the scheduling algorithm.

The simulation scenario is shown in Figure 5.13 where a MN is equipped with two interfaces, namely, WiFi (*if1*) and WiMAX (*if2*) which uses the Time Division Multiplexing model. MN connects to a wireless local domain through *if1* and *if2* simultaneously. There are two access routers, *AR_A* is an 802.11 base station and *AR_B* is an 802.16 base station. Both APs have reserved a buffer of 10 packets for the observation flow. *DGR* is the gateway of the local domain and all of the access routers connect to the Internet via *DGR*. A Corresponding Node *CN* sends traffic to *MN*. The load share of WiFi interface and WiMAX interface is set to 0.5 for both which means that the 2 interfaces are expected to receive the same amount of traffic during the transmission.

5.4.1 Max Flow

This set of simulations measures *Max Flow* which shows how effectively ALBAM, OMS and FLARE can aggregate the bandwidth of the different paths. The larger *Max Flow* is, the higher is the efficiency of the scheduling algorithm. Before measuring the max flow for each algorithm, the single path capacity is measured and the results shown in Figure 5.14. Then, the max flow supported by each scheduling algorithm is measured and the result is shown in Figure 5.15. From Figure 5.15, it can be seen that ALBAM can support the highest flowrate regardless of the TCP versions tested here, namely, Reno, NewReno and Tahoe.

5.4.2 Out-of-Order Ratio

In this section, the ratio of the number of out-of-order packets over the total number of packets transmitted is computed. As TCP is a reliable and sequential connection, the out-of-order packets will trigger erroneous response from the receiver. Hence, the scheduling algorithm should reduce the number of out-of-order packets as much as possible. To show

Figure 5.13: Experiment Scenario

the different scheduling behavior of FLARE when a TCP flow achieves a stable flowrate, the out-of-order packets at the initial state are plotted in Figure 5.16 and those in the stable state are shown in Figure 5.17. From Figure 5.16, it can be seen that the out-of-order ratio of FLARE is higher than ALBAM. It is because FLARE does not consider the buffer overflow issue when TCP congestion window increases rapidly. Hence, at $t = 5s$, there is packet loss due to the limited buffer size under FLARE and the out-of-order ratio increases a lot. However, when TCP flow achieves a stable flowrate, the out-of-order ratio under FLARE decreases rapidly.

Figure 5.14: Single Path Capacity for TCP Flow

Figure 5.17 shows the out-of-order ratio for the three scheduling algorithms for a long period when TCP is in steady state. It can be seen that when flowrate is low (500 kbps), OMS can achieve a good performance for out-of-order packets. However, when the flowrate is high (1000 kbps), OMS's out-of-order packets are more than the other 2 schemes as it uses packet level scheduling rather than flowlet level. Among the 3 scheduling algorithms, ALBAM always performs better than FLARE and OMS.

Figure 5.15: Maximum Flow

5.4.3 Splitting Errors

To evaluate the load-balancing effect of the three scheduling algorithms, the *Splitting Error* (E_S) for each scheduling algorithm is measured. Since FLARE does not switch delivery path when a flow achieves a stable flowrate, only the E_S for FLARE in the unstable state is plotted in Figure 5.18. Although FLARE can achieve a good E_S performance at the initial start of the TCP flow, E_S increases when TCP flow enters the stable state at $t = 10s$. For ALBAM, E_S decreases with simulation time which means that ALBAM can achieve a better E_S performance in the long run under steady state as proven in Figure 5.19 which shows the long-term E_S performance for the three algorithms. From Figure 5.19, it can be seen

Figure 5.16: Out-of-Order Ratio When TCP is in Unstable State

that E_S converges to zero under ALBAM which means that ALBAM can achieve better load balancing than its two counterparts.

5.4.4 Transmission Time

In this section, the time taken for file transmission under the different scheduling algorithms is measured. The transmission time is the duration from the time TCP starts sending a file until the destination receives the entire file. For a given file, the less the transmission time, the better is the performance of the scheduling algorithm. In the experiment, the file is sent with a flowrate of 1000 kbps and the file size varies from 10 KB to 10 MB. Figure 5.20

Figure 5.17: Out-of-Order Ratio When TCP is in Stable State

presents the simulation results. From Figure 5.20, it can be seen that when the file size is small (less than 10 KB), ALBAM can download the file in less time. When the file size is large, ALBAM can yield the same performance as OMS because both utilize two paths simultaneously while FLARE can only use one path when the flow enters the stable state.

5.5 Performance for UDP Flow

Although ALBAM is designed for TCP flow, it can be extended to schedule UDP flow. As UDP does not have acknowledgement mechanism, ALBAM does not employ PNEA to

Figure 5.18: Splitting Error When TCP is in Unstable State

estimate the buffer usage in each path. It inherits the burst scheduling and employs VFD to divide the UDP flow. For each division of the UDP flow, ALBAM chooses one path and sends the whole pack of packets to it. At the same time, ALBAM monitors the transmission delay for each path. If ALBAM switches the delivery path for the UDP flow, it buffers them until the burst interval is safe for the out-of-order packets. The performance of ALBAM for UDP flow is given in the following sections and is compared against OMS. It is to be noted that FLARE is excluded from the comparison as FLARE is focused on TCP flow. The performance metrics used are Max UDP flow that can be aggregated in ALBAM and the amount of out-of-order packets. The experimental setup is the same as TCP flow which is

Figure 5.19: Splitting Error When TCP is in Stable state

presented in Figure 5.13. In the experiment, we let CN sends a UDP flow to MN which is equipped with both WiFi and WiMAX interface. The load share of WiFi interface and WiMAX interface is set to 0.6 : 0.4 because the capacity of WiFi interface and WiMAX interface is 3000kbps and 2100kbps respectively. Figure 5.21 shows the single path capacity for the two interfaces.

5.5.1 Max UDP Flow

When ALBAM is used to schedule UDP flow, both the WiMax and the Wifi interfaces are used to aggregate the bandwidth. The max flow changes according to the VFD interval. The

Figure 5.20: Transmission Time

result is shown in Figure 5.22. From Figure 5.22, it can be seen that if the VFD interval is decreased, the max UDP flow is increased. For example, when the VFD interval is set to 0.2s, the max UDP flow supported by ALBAM is 2900 kbps. If the VFD interval is changed to 0.01s, the max UDP flow is increased to 4200 kbps. However, with the decrease of VFD interval, the out-of-order ratio is increased. Figure 5.22 presents the out-of-order ratio of a 2000 kbps UDP flow under varying VFD interval. Hence, when employing ALBAM for UDP flow, a balance between the capacity and out-of-order packets have to be made.

Figure 5.21: Single Path Capacity for UDP Flow

5.5.2 Out-of-Order Packets for UDP Flow for ALBAM and OMS

In this section, the out-of-order UDP packets for both OMS and ALBAM are measured and shown in Figure 5.23. From Figure 5.23, it can be seen that under each sending rate, ALBAM with a VFD interval equal to 0.01s achieves a lower out-of-order packet ratio than OMS. Hence, for packet-order sensitive UDP applications, ALBAM can give a better performance than OMS.

Figure 5.22: Max UDP Flow and Out-of-Order Ratio vs VFD interval

5.6 Conclusion

Multipath scheduling, which enables a flow to be split among two or more disjoint paths, has many advantages over single path routing, such as load balancing, throughput enhancement and reduced transmission time. This chapter presents a multipath scheduling algorithm, named Adaptive Load Balancing Algorithm (ALBAM) to distribute TCP flows across different wireless paths. Based on flowlet scheduling, ALBAM develops three techniques to improve its performance. Firstly, ALBAM develops a packet number estimation algorithm (PNEA) to predict the buffer usage in each path. With PNEA, ALBAM can greatly reduce if not prevent buffer overflow and schedule the TCP flow to a less congested path before it suffers packet loss. Secondly, ALBAM develops an virtual flowlet division to divide the flow into several virtual flowlets. VFD enables ALBAM to improve the scheduling performance

Figure 5.23: Out-of-Order Ratio for UDP Flow

when the flowlets are missing. Lastly, to obtain more accurate path delay measurement, ALBAM employs the Access Points (AP) to provide statistical delay measurements in each path. Simulation results show that the statistical method can yield a more accurate delay reading than FLARE, especially in one-hop wireless networks. The results also show that ALBAM can provide better performance for TCP connections than its counterparts, namely, FLARE and OMS. ALBAM can be applied for UDP flow as well and it yields better out-of-order ratio than OMS.

Part IV

Conclusion and Future Work

Chapter 6

Conclusion and Future Work

6.1 Conclusion

The research reported in this dissertation focuses mainly on addressing two crucial issues for the integration of heterogeneous networks: mobility management and multipath scheduling.

For mobility management, a two-level mobility framework comprising both local mobility and global mobility has been proposed. The two-level hierarchy aims to handle the two kinds of movement: the movement across networks within a local domain and the movement outside a local domain. Local mobility allows MN to move in a domain without service disruption. An EFLoM has been developed to handle the issue of local mobility management. To enable MN to move across networks within one domain, EFLoM employs three network entities: Mobile Node (MN), Local Anchor Router (LAR) and Wireless Access Gateway (WAG) to reduce the handover delay. Moreover, EFLoM uses an IP header swapping mechanism to avoid the traffic overhead and this improves the network throughput. Compared with other local mobility protocols such as HMIPv6 and PMIPv6, EFLoM has a better performance in terms of handover delay, throughput and signaling cost.

If users move out of one local domain, they are supposed to employ global mobility such as MIP and HIP to handover their sessions. However, global mobility suffers a lot of issues such as triangular routing and long handover latency. An inter-domain mobility protocol, called Traffic Distributor Inter-Domain Mobility (TDIDM), to handle the movement from one domain to its neighboring domain has been proposed. To deliver the cross-domain traffic, TDIDM introduces a new network entity named traffic distributor (TD) in the aggregate network. When MN moves across domain borders, the domain gateway router will notify the TD and the TD will redirect MN's traffic to the new domain. With the help of the TD, TDIDM can interconnect the neighboring local domains and enable mobile nodes to continue their existing sessions when they move across multiple local domains. Compared with other inter-domain mobility protocol such as Neumann's proposal, TDIDM outperforms it in terms of DGR's BCE numbers, transmission delay and handover delay.

Multipath scheduling is another important issue arising from the integration of heterogeneous wireless networks. Compared with single path scheduling, multipath scheduling has many advantages such as load balancing and throughput enhancement. A multipath scheduling algorithm, named Adaptive Load Balancing Algorithm (ALBAM) to distribute TCP flows across different wireless paths has been proposed. Based on flowlet scheduling, ALBAM develops three techniques to improve its performance. Firstly, ALBAM develops a packet number estimation algorithm (PNEA) to predict the buffer usage in each path. With PNEA, ALBAM can greatly reduce if not prevent buffer overflow and schedule the TCP flow to a less congested path before it suffers packet loss. Secondly, ALBAM develops a virtual flowlet division (VFD) to divide the flow into several virtual flowlets. The VFD enables ALBAM to improve the scheduling performance when the flowlets are missing. Lastly, to obtain more accurate path delay measurement, ALBAM employs the Access Points (AP) to provide statistical delay measurements in each path. Simulation results show that the statistical method can yield a more accurate delay reading than FLARE, especially in one-hop

wireless networks. The results also show that ALBAM can provide better performance for TCP connections than its counterparts, namely, FLARE and OMS.

6.2 Future Work

6.2.1 QoS-Enabled Mobility

Future heterogeneous wireless system should not only provide users the ability to move around with their sessions being handed over from one network to another, but also ensure that their Quality-of-Service (QoS) is not degraded during the movement. Therefore, a QoS-enabled mechanism should be integrated into the existing mobility system.

To incorporate the QoS function, a mapping strategy between the different kinds of wireless networks is needed. Owing to the different implementations of wireless technologies, these wireless access network provide different QoS support to users (e.g., bandwidth, latency, power, price, security, reliability, availability). When users make a handoff decision, they need to take the QoS parameter into consideration. Each candidate network can be compared with the current network and the decision is to select one appropriate network which can satisfy the requirement of the users. (e.g., minimum and maximum bandwidth requirement for voice service).

The mapping strategy should not be limited to support the QoS function. Other functions such as available bandwidth estimation and resource sharing are needed to facilitate the implementation of the QoS function. For example, if an application with a high bandwidth requirement is going to be handed over and none of the candidate network can solely provide such a high bandwidth, the QoS mobility can employ multiple networks to combine their bandwidth and satisfy the bandwidth requirement of the application. Hence, after the handover, the application flow may be split and multiple networks have to provide the service

to the users simultaneously.

6.2.2 Integration of Routing, Resource Allocation and Scheduling in Future Wireless Networks

With more and more wireless MAC protocols supporting rate adaption, such as 802.11n and WiMAX, future wireless network is expected to provide users with multi-rate capacities. The multi-rate function helps users overcome the highly volatile nature of the wireless medium and obtain a better performance.

While the rate adaption brings benefits to users, there are some challenges. For multipath scheduling algorithm, the varying rate of the wireless network may result in unexpected congestion. For example, when one receiving interface suffers strong interference, the throughput capacity of the interface may be dropped in order to decrease the Bit Error Rate (BER). However, if multipath scheduling keeps on distributing the same volume of traffic to this low-rate path, congestion may occur. To solve this problem, an integrated routing, resource allocation and scheduling framework which can support both the rate change and the traffic scheduling can be investigated.

In future wireless networks, the new framework can capture and respond quickly to the rate adaption on each link. Upon sensing the rate change, the routing module should estimate the new available throughput and adjust the resource allocation for that path. Then, the routing module performs calculation to change the load weighting between the different paths based on the new throughput and resource. The adjustment of load weighting can help the scheduling algorithm split traffic. Under the cooperation between the routing and the scheduling modules, minimum congestion will be incurred under the dynamic multirate wireless environment.

Author's Publications

Journal Papers

1. **Feng Zhong**, Chai Kiat Yeo, Bu Sung Lee, "Enabling Inter-PMIPv6-domain Handover with Traffic Distributors", Journal of Network and Computer Applications, Volume 33, Issue 4, July 2010, Pages 397-409.
2. **Feng Zhong**, Chai Kiat Yeo, Bu Sung Lee, "Adaptive Load Balancing Algorithm for Multiple Homing Mobile Nodes", Journal of Network and Computer Applications, Volume 35, Issue 1, January 2012, Pages 316-327.
3. **Feng Zhong**, Chai Kiat Yeo, Bu Sung Lee, "EFLoM: An Efficient Framework for Local Mobility", revision has been submitted to Elsevier Computer Communications, 2011.

Conference Papers

1. **Feng Zhong**, Chai Kiat Yeo, Bu Sung Lee, "Adaptive Load Balancing Algorithm for Multi-Homing Mobile Nodes in Local Domain", IEEE Consumer Communications and Networking Conference (CCNC), Las Vegas, USA, 2011.
2. **Feng Zhong**, Shengbo Yang, Chai Kiat Yeo, Bu Sung Lee, "Enabling Inter-PMIPv6-domain Handover with Traffic Distributors", IEEE Consumer Communications and Networking Conference (CCNC), Las Vegas, USA, 2010.
3. **Feng Zhong**, Yang Xia, Chai Kiat Yeo, Bu Sung Lee, Teck Meng Lim, "An Efficient Framework for Local Mobility", IEEE Wireless Communications and Networking Conference (WCNC), Budapest, Hungary, 2009.
4. Shengbo Yang, **Feng Zhong**, Chai Kiat Yeo, Bu Sung Lee, Jeff Boleng, "Position Based Opportunistic Routing for Robust Data Delivery in MANETs", IEEE Global Telecommunications Conference (Globecom), Hawaii, USA, 2009.
5. Chai Kiat Yeo, Bu Sung Lee, **Feng Zhong**, "A mobile SIP: IP mobility support using SIP", International Conference On Ubiquitous Information Management and Communication (ICUMC), Seoul, South Korea, 2008.

Bibliography

- [1] “Ericsson pushes mobile broadband and cloud services,” Feb 2011. [Online]. Available: <http://telecoms.cytalk.com/2011/02/ericsson-pushes-mobile-broadband-and-cloud-services/>
- [2] J. Chen and T. Zhang, *IP-based Next-generation Wireless Networks*. Wiley Online Library, 2004.
- [3] L. Gavrilovska and V. Atanasovski, “Interoperability in future wireless communications systems: A roadmap to 4G,” *Mikrotalesna revija*, vol. 13, no. 1, pp. 19–28, Jan. 2007.
- [4] IEEE, “IEEE Standard for Information Technology - Telecommunications and Information Exchange Between Systems - Local and Metropolitan Area Networks - Specific Requirements. - Part 15.1: Wireless Medium Access Control (MAC) and Physical Layer (PHY) Specifications for Wireless Personal Area Networks (WPANs),” *IEEE Std 802.15.1-2005 (Revision of IEEE Std 802.15.1-2002)*, vol. 1, pp. 1–580, 2005.
- [5] ISO/IEEE, “ISO/IEC Standard for Information Technology - Telecommunications and Information Exchange Between Systems - Local and Metropolitan Area Networks - Specific Requirements Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications (Includes IEEE Std 802.11, 1999 Edition; IEEE Std 802.11A.-1999; IEEE Std 802.11B.-1999; IEEE Std 802.11B.-1999/Cor 1-2001;

and IEEE Std 802.11D.-2001),” *ISO/IEC 8802-11 IEEE Std 802.11 Second edition 2005-08-01 ISO/IEC 8802 11:2005(E) IEEE Std 802.11i-2003 Edition*, vol. 1, pp. 1–721, 2005.

- [6] IEEE, “IEEE Standard for Local and Metropolitan Area Networks - Part 16: Air Interface for Fixed and Mobile Broadband Wireless Access Systems - Amendment for Physical and Medium Access Control Layers for Combined Fixed and Mobile Operation in Licensed Bands,” *802.16e-2005*, vol. 1, pp. 1–502, 2005.
- [7] J. Andrews, A. Ghosh, and R. Muhamed, *Fundamentals of WiMAX: Understanding Broadband Wireless Networking*. Prentice Hall, 2007.
- [8] L. Nuaymi, *WiMAX: Technology for Broadband Wireless Access*. Wiley, 2007.
- [9] J. Eberspächer and H. Vögel, *GSM Global System for Mobile Communication*. Teubner, 1999.
- [10] H. Holma, *HSDPA/HSUPA for UMTS: High Speed Radio Access for Mobile Communications*. Wiley, 2006.
- [11] M. Buddhikot, G. Chandranmenon, S. Han, Y. Lee, S. Miller, and L. Salgarelli, “Design and implementation of a WLAN/CDMA2000 interworking architecture,” *IEEE Commun. Mag.*, vol. 41, no. 11, pp. 90–100, Nov. 2003.
- [12] F. Marquez, M. Rodriguez, T. Valladares, T. De Miguel, and L. Galindo, “Interworking of IP multimedia core networks between 3GPP and WLAN,” *IEEE Wireless Commun. Mag.*, vol. 12, no. 3, pp. 58–65, Mar. 2005.
- [13] J. Kang and B. Nath, “Adaptive QoS control by toggling voice traffic between circuit and packet cellular networks,” in *Proc. IEEE Global Telecommun. Conf. (GLOBE-COM’03)*, Dec. 2003, pp. 3498–3503.

- [14] T. Nomura, H. Ueda, C. Itoh, H. Kurokawa, T. Tsuboi, and H. Kasai, "Design of optical switching module for gigabit ethernet optical switched access network," *IEICE TRANSACTIONS ON COMMUNICATIONS E SERIES B*, vol. 89, pp. 3021–3031, 2006.
- [15] A. S. Diwan, R. Guerin, and K. N. Sivarajan, "Performance analysis of speeded-up high-speed packet switches," *JOURNAL OF HIGH SPEED NETWORKS*, vol. 10, pp. 161–186, 2001.
- [16] V. W. S. Chan, K. L. Hall, E. Modiano, and K. A. Rauschenback, "Architectures and technologies for high-speed optical data networks," *JOURNAL OF LIGHTWAVE TECHNOLOGY LT*, vol. 16, pp. 2146–2168, 1998.
- [17] D. Cavalcanti, D. Agrawal, C. Cordeiro, B. Xie, and A. Kumar, "Issues in integrating cellular networks WLANs and MANETs: A futuristic heterogeneous wireless network," *IEEE Wireless Commun. Mag.*, vol. 12, no. 3, pp. 30–41, Jun. 2005.
- [18] G. Ruggeri, A. Iera, and S. Polito, "802.11-based Wireless-LAN and UMTS interworking: Requirements, proposed solutions and open issues," *Elsevier Comput. Netw.*, vol. 47, no. 2, pp. 151–166, Feb. 2005.
- [19] A. Salkintzis, "Interworking techniques and architectures for WLAN/3G integration toward 4G mobile data networks," *IEEE Wireless Commun. Mag.*, vol. 11, no. 3, pp. 50–61, Jun. 2004.
- [20] J. Pan, R. Jain, S. Paul, and C. So-in, "MILSA: A new evolutionary architecture for scalability, mobility, and multihoming in the future internet," *IEEE J. Sel. Areas Commun.*, vol. 28, no. 8, pp. 1344–1362, Aug. 2010.

- [21] J. Song, S. Lee, and D. Cho, "Hybrid coupling scheme for UMTS and Wireless LAN interworking," in *Proc. IEEE Vehicular Tech. Conf. (VTC'03-Fall)*, Dec. 2003, pp. 2247–2251.
- [22] S. Tsao and C. Lin, "Design and evaluation of UMTS-WLAN interworking strategies," in *Proc. IEEE Vehicular Tech. Conf. (VTC'02-Fall)*, Dec. 2002, pp. 777–781.
- [23] M. Yabusaki, Y. Okumura, and M. Tamura, "Next mobile network architecture," in *Proc. IEEE Int. Conf. Commun. (ICC'10)*, Jun. 2010, pp. 1–5.
- [24] P. Chan, E. Lo, R. Wang, E. Au, V. Lau, R. Cheng, W. Mow, R. Murch, and K. Letaief, "The evolution path of 4G networks: FDD or TDD?" *IEEE Commun. Mag.*, vol. 44, no. 12, pp. 42–50, Dec. 2006.
- [25] M. Rumney, *LTE and the evolution to 4G Wireless: Design and Measurement Challenges*. Wiley, 2009.
- [26] K. Ahmavaara, H. Haverinen, and R. Pichna, "Interworking architecture between 3GPP and WLAN systems," *IEEE Commun. Mag.*, vol. 41, no. 11, pp. 74–81, Nov. 2003.
- [27] O. Falowo and H. Chan, "AAA and mobility management in UMTSWLAN interworking," in *Proc. Int. Conf. Telecommun (ICT'05)*, 2005.
- [28] C. Yang, K. Chu, and Y. Yang, "3G and WLAN interworking security: Current status and key issues," *Int J. Netw. Security*, vol. 2, no. 1, pp. 1–13, Jan. 2006.
- [29] G. Koien and T. Haslestad, "Security aspects of 3G-WLAN interworking," *IEEE Commun. Mag.*, vol. 41, no. 11, pp. 82–88, Nov. 2003.

- [30] L. Zhang, W. Jia, S. Wen, and D. Yao, "A man-in-the-middle attack on 3G-WLAN interworking," in *Proc. IEEE Int. Conf. Communications and Mobile Computing (CM-C'10)*, Apr. 2010, pp. 121–125.
- [31] P. Kyasanur and N. Vaidya, "Routing and interface assignment in multi-channel multi-interface wireless networks," in *Proc. IEEE Wireless Commun. and Netw. Conf. (WCNC'05)*, Mar. 2005, pp. 2051–2056.
- [32] P. Bahl, A. Adya, J. Padhye, and A. Walman, "Reconsidering wireless systems with multiple radios," *ACM SIGCOMM Comput. Commun. Rev.*, vol. 34, no. 5, pp. 39–46, Oct. 2004.
- [33] M. Alicherry, R. Bhatia, and L. Li, "Joint channel assignment and routing for throughput optimization in multi-radio wireless mesh networks," in *Proc. ACM Int. Conf. Mobile Comput. and Netw. (MobiCom'05)*, Aug. 2005, pp. 58–72.
- [34] A. Attar and H. Aghvami, "A framework for unified spectrum management (USM) in heterogeneous wireless networks," *IEEE Commun. Mag.*, vol. 45, no. 9, pp. 44–51, Sep. 2007.
- [35] A. Hasib and A. Fapojuwo, "Joint radio resource management over very tightly coupled heterogeneous networks for multimode reconfigurable terminals," in *Proc. IEEE Vehicular Tech. Conf. (VTC'06-Spring)*, Aug. 2006, pp. 727–731.
- [36] M. Dillinger, K. Madani, and N. Alonistioti, *Software Defined Radio: Architectures, Systems, and Functions*. John Wiley & Sons Inc, 2003.
- [37] N. Sattari, P. Pangalos, and H. Aghvami, "Seamless handover between WLAN and UMTS," in *Proc. IEEE Vehicular Tech. Conf. (VTC'04-Spring)*, Aug. 2004, pp. 3035–3038.

- [38] H. Choi, O. Song, and D. Cho, "A seamless handoff scheme for UMTS-WLAN inter-working," in *Proc. IEEE Global Telecommun. Conf. (GLOBECOM'04)*, Dec. 2004, pp. 1559–1564.
- [39] S. Paul, R. Yates, D. Raychaudhuri, and J. Kurose, "The cache-and-forward network architecture for efficient mobile content delivery services in the future internet," in *Proc. IEEE Innovations in NGN: Future Network and Services (K-INGN'08)*, May 2008, pp. 367–374.
- [40] C. Guo, Z. Guo, Q. Zhang, and W. Zhu, "A seamless and proactive end-to-end mobility solution for roaming across heterogeneous wireless networks," *IEEE J. Sel. Areas Commun.*, vol. 22, no. 5, pp. 834–848, May 2004.
- [41] A. Hasswa, N. Nasser, and H. Hossanein, "Generic vertical handoff decision function for heterogeneous wireless," in *Proc. IEEE Int. Conf. Wireless and Optical Commun. Netw. (WOCN'05)*, Mar. 2005, pp. 239–243.
- [42] G. Lampropoulos, N. Passas, L. Merakos, and A. Kaloxylos, "Handover management architectures in integrated WLAN/Cellular networks," *IEEE Commun. Surveys Tuts.*, vol. 7, no. 4, pp. 30–44, Feb. 2005.
- [43] C. Boyaci, B. Li, and Y. Xia, "An investigation on the nature of wireless scheduling," in *Proc. IEEE Int. Conf. Comput. Commun. (INFOCOM'10)*, Mar. 2010, pp. 1–9.
- [44] M. Neely, E. Modiano, and C. Li, "Fairness and optimal stochastic control for heterogeneous networks," in *Proc. IEEE Int. Conf. Comput. Commun. (INFOCOM'05)*, Mar. 2005, pp. 1723–1734.
- [45] W. Song, W. Zhuang, and Y. Cheng, "Load balancing for Cellular/WLAN integrated networks," *IEEE Netw.*, vol. 21, no. 1, pp. 27–33, Jan. 2007.

- [46] C. Perkins, S. Alpert, and B. Woolf, *Mobile IP: Design Principles and Practices*. Addison-Wesley Longman Publishing Co., Inc., 1997.
- [47] C. Perkins, “Mobile IP,” *IEEE Commun. Mag.*, vol. 35, no. 5, pp. 84–99, May 1997.
- [48] —, “Mobile networking through mobile IP,” *IEEE Internet Comput.*, vol. 2, no. 1, pp. 58–69, Jan. 1998.
- [49] H. Soliman, “Mobile IPv6 support for dual stack hosts and routers,” RFC 5555 (Proposed Standard), Internet Engineering Task Force, Jun. 2009.
- [50] R. Moskowitz, P. Nikander, P. Jokela, and T. Henderson, “Host identity protocol,” RFC 5201 (Experimental), Internet Engineering Task Force, Apr. 2008.
- [51] T. Heer and S. Varjonen, “Host identity protocol certificates,” RFC 6253 (Experimental), Internet Engineering Task Force, May 2011.
- [52] T. Lim, C. Yeo, F. Lee, and Q. Le, “TMSP: Terminal mobility support protocol,” *IEEE Trans. Mobile Comput.*, vol. 8, no. 6, pp. 849–863, Jun. 2009.
- [53] P. Nikander, T. Henderson, C. Vogt, and J. Arkko, “End-host mobility and multihoming with the host identity protocol,” RFC 5206 (Experimental), Internet Engineering Task Force, Apr. 2008.
- [54] P. Nikander and J. Laganier, “Host identity protocol (HIP) domain name system (DNS) extensions,” RFC 5205 (Experimental), Internet Engineering Task Force, Apr. 2008.
- [55] J. Rosenberg, H. Schulzrinne, G. Camarillo, A. Johnston, J. Peterson, R. Sparks, M. Handley, and E. Schooler, “SIP: Session initiation protocol,” RFC 3261 (Proposed Standard), Internet Engineering Task Force, Jun. 2002.

- [56] H. Schulzrinne and E. Wedlund, "Application-layer mobility using SIP," in *Proc. IEEE Service Portability and Virtual Customer Environments*, Dec. 2000, pp. 29–36.
- [57] E. Wedlund and H. Schulzrinne, "Mobility support using SIP," in *Proc. ACM Int. Workshop Wireless mobile multimedia (WoWMoM'99)*, Jun. 1999, pp. 76–82.
- [58] D. Saha, A. Mukherjee, I. Misra, and M. Chakraborty, "Mobility support in IP: A survey of related protocols," *IEEE Netw.*, vol. 18, no. 6, pp. 34–40, Jun. 2004.
- [59] I. Akyildiz, J. Xie, and S. Mohanty, "A survey of mobility management in next-generation All-IP-based wireless systems," *IEEE Wireless Commun. Mag.*, vol. 11, no. 4, pp. 16–28, Apr. 2004.
- [60] D. Le, X. Fu, D. Hogrefe *et al.*, "A review of mobility support paradigms for the Internet," *IEEE Commun. Surveys Tuts.*, vol. 8, no. 1, pp. 2–15, Jan. 2006.
- [61] J. Kempf, "Problem statement for network-based localized mobility management (NETLMM)," RFC 4830 (Informational), Internet Engineering Task Force, Apr. 2007.
- [62] R. Koodli, "IP address location privacy and Mobile IPv6: Problem statement," Feb. 2007.
- [63] A. Campbell, J. Gomez, S. Kim, C. Wan, Z. Turanyi, and A. Valkó, "Comparison of IP micromobility protocols," *IEEE Wireless Commun. Mag.*, vol. 9, no. 1, pp. 72–82, Jan. 2002.
- [64] A. Campbell and J. Gomez-Castellanos, "IP micro-mobility protocols," *ACM SIG-MOBILE Mobile Comput. and Commun. Rev.*, vol. 4, no. 4, pp. 45–53, Oct. 2000.
- [65] P. Reinbold and O. Bonaventure, "IP micro-mobility protocols," *IEEE Commun. Surveys Tuts.*, vol. 5, no. 1, pp. 40–57, Jul. 2003.

- [66] A. Valkó, “Cellular IP: A new approach to internet host mobility,” *ACM SIGCOMM Comput. Commun. Rev.*, vol. 29, no. 1, pp. 50–65, Jan. 1999.
- [67] R. Ramjee, K. Varadhan, L. Salgarelli, S. Thuel, S. Wang, and T. La Porta, “HAWAII: A domain-based approach for supporting mobility in wide-area wireless networks,” *IEEE/ACM Trans. Netw.*, vol. 10, no. 3, pp. 396–410, Jun. 2002.
- [68] H. Soliman, C. Castelluccia, K. ElMalki, and L. Bellier, “Hierarchical mobile IPv6 (HMIPv6) mobility management,” RFC 5380 (Proposed Standard), Internet Engineering Task Force, Oct. 2008.
- [69] S. Gundavelli, K. Leung, V. Devarapalli, K. Chowdhury, and B. Patil, “Proxy mobile IPv6,” RFC 5213 (Proposed Standard), Internet Engineering Task Force, Aug. 2008.
- [70] A. Helmy, M. Jaseemuddin, and G. Bhaskara, “Multicast-based mobility: A novel architecture for efficient micromobility,” *IEEE J. Sel. Areas Commun.*, vol. 22, no. 4, pp. 677–690, Apr. 2004.
- [71] J. Kempf, “Goals for network-based localized mobility management (NETLMM),” RFC 4831 (Informational), Internet Engineering Task Force, Apr. 2007.
- [72] C. Vogt and J. Kempf, “Security threats to network-based localized mobility management (NETLMM),” RFC 4832 (Informational), Internet Engineering Task Force, Apr. 2007.
- [73] Y. Gvovon, J. Kempf, and A. Yegin, “Scalability and robustness analysis of mobile IPv6, fast mobile IPv6, hierarchical mobile IPv6, and hybrid IPv6 mobility protocols using a large-scale simulation,” in *Proc. IEEE Int. Conf. Commun. (ICC’04)*, Jun. 2004, pp. 4087–4091.
- [74] X. Pérez-Costa, M. Torrent-Moreno, and H. Hartenstein, “A performance comparison of Mobile IPv6, hierarchical Mobile IPv6, fast handovers for Mobile IPv6 and their

combination,” *ACM SIGCOMM Comput. Commun. Rev.*, vol. 7, no. 4, pp. 5–19, Oct. 2003.

[75] G. Kim, *Low Latency Cross Layer Handover Scheme in Proxy Mobile IPv6 Domain*. Springer, 2008.

[76] J. Lei and X. Fu, “Evaluating the benefits of introducing PMIPv6 for localized mobility management,” in *Proc. IEEE Int. Wireless Commun and Mobile Comput Conf. (IWCMC’08)*, Aug. 2008, pp. 74–80.

[77] N. Neumann, J. Lei, X. Fu, and G. Zhang, “I-PMIP: An inter-domain mobility extension for Proxy-Mobile IP,” in *Proc. IEEE Wireless Commun. and Netw. Conf. (WCNC’09)*, Mar. 2009, pp. 994–999.

[78] N. Neumann, X. Fu, J. Lei, and G. Zhang, “Inter-domain handover and data forwarding between Proxy Mobile IPv6 domains,” Internet Engineering Task Force, 2009.

[79] C. Perkins and D. Johnson, “Route optimization for mobile IP,” *Springer Cluster Comput.*, vol. 1, no. 2, pp. 161–176, Jun. 1998.

[80] H. Huang, J. Cai, A. Kassler, and C. Fu, “Load-sharing in wireless multi-homed systems,” in *Proc. IEEE Int. Conf. Commun. (ICC’05)*, Jun. 2005, pp. 3489–3493.

[81] J. Iyengar, P. Amer, and R. Stewart, “Concurrent multipath transfer using SCTP multihoming over independent end-to-end paths,” *IEEE/ACM Trans. Netw.*, vol. 14, no. 5, pp. 951–964, Oct. 2006.

[82] —, “Retransmission policies for concurrent multipath transfer using SCTP multihoming,” in *Proc. IEEE Int. Conf. Netw. (ICON’04)*, Sep. 2004, pp. 713–719.

[83] M. Tsai, N. Chilamkurti, J. Park, and C. Shieh, “Multi-path transmission control scheme combining bandwidth aggregation and packet scheduling for real-time stream-

ing in multi-path environment,” *IEEE Commun. IET*, vol. 4, no. 8, pp. 937–945, May 2010.

- [84] L. Magalhaes and R. Kravets, “Transport level mechanisms for bandwidth aggregation on mobile hosts,” in *Proc. IEEE Int. Conf. Netw. Protocols (ICNP’01)*, Nov. 2001, pp. 165–171.
- [85] H. Hsieh and R. Sivakumar, “pTCP: An end-to-end transport layer protocol for striped connections,” in *Proc. IEEE Int. Conf. Netw. Protocols (ICNP’02)*, Nov. 2002, pp. 24–33.
- [86] A. A. E. Al, T. Saadawi, and M. Lee, “LS-SCTP: A bandwidth aggregation technique for stream control transmission protocol,” *Elsevier Comput. Commun.*, vol. 27, no. 10, pp. 1012–1024, Jun. 2004.
- [87] C. Cetinkaya and E. Knightly, “Opportunistic traffic scheduling over multiple network paths,” in *Proc. IEEE Int. Conf. Comput. Commun. (INFOCOM’04)*, Mar. 2004, pp. 1928–1937.
- [88] C. Cetinkaya, “Improving the efficiency of multipath traffic via opportunistic traffic scheduling,” *Elsevier Comput. Netw.*, vol. 51, no. 8, pp. 2181–2197, Jun. 2007.
- [89] S. Kandula, D. Katabi, S. Sinha, and A. Berger, “Dynamic load balancing without packet reordering,” *ACM SIGCOMM Comput. Commun. Rev.*, vol. 37, no. 2, pp. 51–62, Apr. 2007.
- [90] S. Sinha, S. Kandula, and D. Katabi, “Harnessing TCP’s burstiness using flowlet switching,” in *Proc. SIGCOMM Workshop Hot Topics in Netw. (HotNets’04)*, 2004.
- [91] T. Lim, J. Tantra, B. Lee, C. Yeo, and Y. Xia, “Terminal-assisted network mobility management,” in *Proc. IEEE Wireless Commun. and Netw. Conf. (WCNC’07)*, Mar. 2007, pp. 3664–3669.

- [92] T. Lim, B. Lee, C. Yeo, J. Tantra, and Y. Xia, "A terminal-assisted route optimized NEMO management," *Springer Telecommun. Systems*, vol. 42, no. 3, pp. 263–272, Dec. 2009.
- [93] D. Johnson, C. Perkins, and J. Arkko, "Mobility support in IPv6," RFC 3775 (Proposed Standard), Internet Engineering Task Force, Jun. 2004.
- [94] R. Droms, J. Bound, B. Volz, T. Lemon, C. Perkins, and M. Carney, "Dynamic host configuration protocol for ipv6 (DHCPv6)," RFC 3315 (Proposed Standard), Internet Engineering Task Force, Jul. 2003.
- [95] K. Chiang and N. Shenoy, "A 2-D random-walk mobility model for location-management studies in wireless networks," *IEEE Trans. Veh. Technol.*, vol. 53, no. 2, pp. 413–424, Feb. 2004.
- [96] T. Narten, E. Nordmark, W. Simpson, and H. Soliman, "Neighbor discovery for IP version 6 (IPv6)," RFC 4861 (Draft Standard), Internet Engineering Task Force, Sep. 2007.
- [97] S. Thomson, T. Narten, and T. Jinmei, "IPv6 stateless address autoconfiguration," RFC 4862 (Draft Standard), Internet Engineering Task Force, Sep. 2007.
- [98] P. McCann and T. Hiller, "An internet infrastructure for cellular CDMA networks using Mobile IP," *IEEE Personal Commun. Mag.*, vol. 7, no. 4, pp. 26–32, Aug. 2000.
- [99] G. Patel and S. Dennett, "The 3GPP and 3GPP2 movements toward an All-IP mobile network," *IEEE Personal Commun. Mag.*, vol. 7, no. 4, pp. 62–64, Aug. 2000.
- [100] G. Carneiro, J. Ruela, and M. Ricardo, "Cross-layer design in 4G wireless terminals," *IEEE Wireless Commun. Mag.*, vol. 11, no. 2, pp. 7–13, Apr. 2004.

- [101] Y. Kim, B. Jeong, J. Chung, C. Hwang, J. Ryu, K. Kim, and Y. Kim, "Beyond 3G: Vision, requirements, and enabling technologies," *IEEE Commun. Mag.*, vol. 41, no. 3, pp. 120–124, Mar. 2003.
- [102] V. Jacobson, "Congestion avoidance and control," *ACM SIGCOMM Comput. Commun. Rev.*, vol. 18, no. 4, pp. 314–329, Aug. 1988.
- [103] H. Balakrishnan, V. Padmanabhan, S. Seshan, and R. Katz, "A comparison of mechanisms for improving TCP performance over wireless links," *IEEE/ACM Trans. Netw.*, vol. 5, no. 6, pp. 756–769, Jun. 1997.
- [104] K. Fall and S. Floyd, "Simulation-based comparisons of Tahoe, Reno, and Sack TCP," *ACM SIGCOMM Comput. Commun. Rev.*, vol. 26, no. 3, pp. 5–21, Jul. 1996.
- [105] S. Xu, T. Saadawi, and M. Lee, "Comparison of TCP Reno and Vegas in wireless mobile ad hoc networks," in *Proc. IEEE Annual Conf. on Local Computer Networks (LCN'00)*, Nov. 2000, pp. 42–43.
- [106] J. Mo, R. La, V. Anantharam, and J. Walrand, "Analysis and comparison of TCP Reno and Vegas," in *Proc. IEEE Int. Conf. Comput. Commun. (INFOCOM'99)*, Mar. 1999, pp. 1556–1563.
- [107] W. Andi, Y. Xia, C. Yeo, and B. Lee, "An interface management framework for multihomed terminal," in *Proc. IEEE Int. Conf. Inform. Commun. and Signal Processing (ICICS'09)*, Dec. 2009, pp. 1–5.
- [108] —, "MIH based SIP mobility management scheme in heterogeneous wireless networks," in *Proc. IEEE Int. Conf. Mobile Ubiquitous Comput. Systems, Services and Tech. (UBICOMM'08)*, Oct. 2008, pp. 7–12.
- [109] Y. Tian, K. Xu, and N. Ansari, "TCP in wireless environments: Problems and solutions," *IEEE Commun. Mag.*, vol. 43, no. 3, pp. S27–S32, Mar. 2005.

- [110] G. Xylomenos, G. Polyzos, P. Mahonen, and M. Saaranen, "TCP performance issues over wireless links," *IEEE Commun. Mag.*, vol. 39, no. 4, pp. 52–58, Apr. 2001.
- [111] M. Chan and R. Ramjee, "Improving TCP/IP performance over third-generation wireless networks," *IEEE Trans. Mobile Comput.*, vol. 7, no. 4, pp. 430–443, Apr. 2007.