

AN ALTERNATIVE ROUTE OPTIMIZATION HANDOFF SCHEME  
FOR MOBILE IPV6

HISHAM MOSTAFA ATTIR

A project report submitted in partial fulfillment of the  
requirements for the award of the degree of  
Master of Science (Computer Science)

Faculty of Computer Science and Information Systems  
Universiti Teknologi Malaysia

NOVEMBER 2008

This thesis is dedicated to my beloved family and friends.

I would also like to dedicate this thesis to the late Ms. Nurul Haini Bt. Anuar  
may Allah bless her soul with his Mercy and Forgiveness.

## **ACKNOWLEDGEMENT**

I would like to grab this opportunity to thank my supervisor, Dr. Shukor Bin Abd. Razak and my co-supervisor Prof. Dr. Norsheila Bt. Faisal for attention, encouragement and guidance throughout the course of this study.

Sincere thanks and gratitude to my beloved family for all supports and understandings they have given to me. I am grateful to all my colleagues, friends, staff, and lecturers in Faculty of Computer Science and Information Systems, Universiti Teknologi Malaysia.

## ABSTRACT

Mobile nodes using MIPv6 suffer long handoff delays due to the Route Optimization registration process. Route Optimization eliminates the triangular routing effect faced in MIPv4. The long handoff latencies may cause severe performance degradation of transmission protocols such as TCP and UDP. Such negative effect can be reduced by some enhancements to the handoff procedure. FastRA, RA Caching, Advanced DAD, Optimistic DAD and Optimistic Node are some of the handoff schemes proposed to reduce the handoff latencies. The aim of this project is to propose an alternative approach to reduce the handoff latencies in MIPv6. Early Handoff (EH) is a handoff scheme that takes advantage of the coverage overlapping areas to perform earlier and smoother handoffs. Simulations using NS-2 and Mobiwan had been carried out to evaluate the performances of EH, FastRA and Optimistic Node. The simulations were conducted within addressing and security issues free environments using 1, 10, 20 and 30 Mobile Nodes settings. The results show that EH has achieved good results in terms of low handoff latency, low packets loss rate, low signaling load and good bandwidth per station when implemented in environments with little number of nodes. On the other hand, FastRA outperformed Optimistic Nodes and EH respectively when the number of Mobile Nodes exceeded 20 nodes.

## ABSTRAK

Nod bergerak yang menggunakan MIPv6 mempunyai kelemahan iaitu masa menunggu yang panjang ketika sambungan disebabkan oleh proses pendaftaran laluan optimum (Route Optimization). Route Optimization dapat mengatasi masalah laluan segitiga yang dihadapi dalam MIPv4. Masalah sambungan yang panjang boleh menyebabkan penurunan prestasi yang besar terhadap protokol transmisi seperti TCP dan UDP. Masalah ini boleh dikurangkan dengan melakukan beberapa pengubahsuaian pada prosidur sambungan. FastRA, RA Caching, Advanced DAD, Optimistic DAD dan Optimistic Node adalah antara beberapa skema sambungan yang telah dicadangkan untuk mengurangkan masalah yang dinyatakan di atas. Matlamat projek ini adalah untuk mencadangkan satu pendekatan alternatif untuk mengurangkan masalah tersembunyi sambungan MIPv6. Sambungan terawal (EH) ialah satu skema sambungan yang mengambil kelebihan daripada penguasaan kawasan-kawasan yang bertindih untuk melaksanakan sambungan dengan lebih awal dan lancar. Simulasi-simulasi menggunakan NS-2 dan Mobiwan telah dijalankan untuk menilai prestasi EH, FastRA dan Optimistic Node. Simulasi-simulasi tersebut telah dikendalikan dalam isu-isu keselamatan persekitaran bebas menggunakan 1, 10, 20 dan 30 set nod bergerak. Hasil dari simulasi menunjukkan bahawa EH telah mencapai hasil yang baik dalam terma masalah tersembunyi sambungan yang rendah, kadar kehilangan paket yang rendah, beban isyarat yang rendah dan jalur lebar yang baik bagi setiap perhentian apabila dilaksanakan di dalam persekitaran dengan jumlah nod yang sedikit. Di dalam situasi yang lain, FastRA menunjukkan prestasi yang lebih baik berbanding Optimistic Nodes dan EH apabila jumlah nod yang bergerak melebihi 20 nod.

## TABLE OF CONTENTS

CHAPTER	TITLE	PAGE
	<b>PROJECT TITLE</b>	i
	<b>DECLARATION</b>	ii
	<b>DEDICATION</b>	iii
	<b>ACKNOWLEDGMENT</b>	iv
	<b>ABSTRACT</b>	v
	<b>ABSTRAK</b>	vi
	<b>TABLE OF CONTENTS</b>	vii
	<b>LIST OF TABLES</b>	xi
	<b>LIST OF FIGURES</b>	xii
	<b>LIST OF ABBREVIATIONS</b>	xiv
	<b>LIST OF APPENDICES</b>	xvi
<b>1</b>	<b>INTRODUCTION</b>	<b>1</b>
	1.1 Introduction	1
	1.2 Networking Technologies	2
	1.2.1 IPv4	2
	1.2.2 IPv6	3
	1.2.3 Wireless Technology	3
	1.3 Problem Background	4
	1.4 Problem Statement	6

1.5 Project Objective	7
1.6 Project Scope	7
1.7 Importance of Study	8
1.8 Organization of the Report	8
1.9 Summary	10
<b>2 LITERATURE REVIEW</b>	<b>11</b>
2.1 Introduction	11
2.2 The TCP/IP Protocol Suite	12
2.2.1 The TCP Protocol Concept	14
2.2.2 The UDP Protocol Concept	16
2.3 Mobile IPv6	18
2.3.1 Route Optimization	20
2.4 Handoff Procedure in MIPv6	21
2.4.1 Router Discovery	22
2.4.1.1 FastRA	24
2.4.1.2 RA Caching	25
2.4.2 Movement Detection	26
2.4.3 Address Configuration	27
2.4.3.1 Duplicate Address Detection	28
2.4.3.2 Alternative Address Configuration Methods	29
A) Advance Duplicate Address Detection	29
B) MLD-DAD	30
C) Optimistic DAD	30
2.4.4 Home Agent and Corresponding Node Registration	32
2.4.4.1 Mobile IPv6 Optimization	33
2.5 Comparison between Existing Handoff Schemes	36
2.5.1 Fast RA	36
2.5.2 RA Caching	36

2.5.3	Advanced DAD	37
2.5.4	Optimistic DAD	37
2.5.5	Optimistic Mobile Node	37
2.6	Simulation Tools	39
2.6.1	Network Simulator 2	39
2.6.2	MobiWan: ns-2 Extension to Support mobility in IPv6 Networks	40
2.7	Summary	40
<b>3</b>	<b>METHODOLOGY AND FRAMEWORK</b>	<b>42</b>
3.1	Introduction	42
3.2	Operational Framework	43
3.2.1	Phase One	43
3.2.2	Phase Two	44
3.2.3	Phase Three	44
3.2.4	Phase Four	45
3.2.5	Phase Five	45
3.2.6	Phase Six	46
3.3	The Network Model	47
3.4	The Simulation Model	48
3.5	The Simulation Setup	49
3.5.1	Simulation Metrics	52
3.6	Simulation Test-Bed	53
3.7	Summary	53
<b>4</b>	<b>EARLY HANDOFF (EH)</b>	<b>55</b>
4.1	Introduction	55
4.2	Early Handoff	56
4.3	EH Implementation	59
4.3.1	Router Advisements Beacon	59
4.3.1	Adding the EH Code	59

4.4 Summary	62
<b>5 SIMULATION RESULTS</b>	<b>64</b>
5.1 Introduction	64
5.2 Simulation Results	65
5.2.1 Handoff Latency	65
5.2.2 Packet Loss	70
5.2.3 Signaling Load	72
5.2.4 Bandwidth per Station	73
5.3 Discussion	75
5.4 Summary	77
<b>6 ACHIEVEMENTS, LINITATIONS AND FUTURE RECOMMENDATIONS</b>	<b>78</b>
6.1 Introduction	78
6.2 Achievements of the Study	79
6.3 Limitations	82
6.4 Recommendations for Future Work	82
<b>REFERENCES</b>	<b>84</b>
<b>Appendices A – E</b>	<b>88 - 98</b>

**LIST OF TABLES**

<b>TABLE NO.</b>	<b>TITLE</b>	<b>PAGE</b>
2.1	The Differences in Techniques used in Existing Handoff Schemes and their Respective Advantages and Limitations.	38
3.1	The Simulation Scenarios.	51
3.2	Simulation Measurement Metrics.	53
5.1	Criteria of Measurement.	75
5.2	Single Node Scenario Performance Comparison Results.	75
5.3	Ten Nodes Scenario Performance Comparison Results.	76
5.4	Twenty Nodes Scenario Performance Comparison Results.	76
5.5	Thirty Nodes Scenario Performance Comparison Results.	76

## LIST OF FIGURES

FIGURE	TITLE	PAGE
1.1	IPv6 Wireless Network Architecture.	4
1.2	Standard MIPv6 Handoff Procedure.	5
2.1	TCP/IP Suite Protocol Layers.	13
2.2	The TCP Packet Header.	14
2.3(a)	Acknowledgment Policies Immediate	16
2.3(b)	Acknowledgment Policies Cumulative.	16
2.4	The UDP Packet.	17
2.5	Home and Corresponding Registrations.	19
2.6	Tunneling in MIPv4.	20
2.7	Route Optimization in MIPv6.	21
2.8	Unicast and Multicast Router Advertisement Messages in Standard Mode.	23
2.9	Unicast and Multicast Router Advertisement Messages in FastRA Mode.	24
2.10	Optimistic Mobile Node.	35
3.1	Operational Framework.	46
3.2	Simulation Network Model.	47
3.3	TCL Code for Generating FTP over TCP Traffic in NS-2.	50
3.4	TCL Code for Generating CBR over UDP Traffic in NS-2.	50

3.5	Simulation Model.	52
4.1	Early Handoff Scheme Flowchart.	58
4.2	Set BS Beacon to 500ms.	59
4.3	The Method Where the Code Will Be Added.	60
4.4	Set a Pointer to the Second BS in the BS List.	60
4.5	Router Advertisement Timer.	60
4.6	Checking If the Timer Is More than 1 Second.	60
4.7	A New Router Advertisement Has Been Received.	61
4.8	Method for Changing the Registration Priorities.	62
4.9	Registering a MN with CN then HA.	62
5.1	TCP Packets Interruption during Handoff.	66
5.2	UDP Packets Interruption during Handoff.	67
5.3(a)	TCP Interruption in EH.	68
5.3(b)	TCP Interruption in FastRA and Optimistic Node.	68
5.4(a)	UDP Interruption in EH.	68
5.4(b)	UDP Interruption in FastRA and Optimistic Node.	68
5.5	Impact of Number of MN on the Handoff Latency.	69
5.6(a)	Packet Loss Rates in TCP Related to Number of Nodes.	71
5.6(b)	Packet Loss Rates in UDP Related to Number of Nodes.	71
5.7	Signaling Load Vs Number of Nodes.	73
5.8	Bandwidth per Station.	74

**LIST OF ABBREVIATIONS**

A-DAD	-	Advance Duplicate Address Detection
AP	-	Access Point
AR	-	Access-point Router
BA	-	Binding Acknowledgment
BU	-	Binding Update
CN	-	Corresponding Node
CoA	-	Care-of-Address
CR	-	Central Router
DAD	-	Duplication Address Detection
DHCPv6	-	Dynamic Host Configuration Protocol for IPv6
DNS	-	Domain Name Service
FA	-	Foreign Agent
FastRA	-	Fast Router Advertisement
FDDI	-	Fiber Distributed Data Interface
FTP	-	File Transfer Protocol
HA	-	Home Agent
HTTP	-	Hyper Text Transfer Protocol
ICMP	-	Internet Control Message Protocol
IETF	-	Internet Engineering Task Force
IPv4	-	Internet Protocol version 4
IPv6	-	Internet Protocol version 6
LAN	-	Local Area Network
LD	-	Link Delay
MAC	-	Media Access Control

MIPv4	-	Mobile Internet Protocol version 4
MIPv6	-	Mobile Internet Protocol version 6
MLD	-	Multicast Listener Discovery report
MLR	-	Multicast Listener Report message
MN	-	Mobile Node
NAT	-	Network Address Translation
ND	-	Neighbor Discovery
NIC	-	Network Interface Card
NS	-	Neighbor Solicitation message
ns2	-	Network Simulator 2
OSI	-	Open System Interconnection
POTS	-	Plain Old Telephone Service
RA	-	Router Advertisement message
RFC	-	Request For Comments
RO	-	Route Optimization
RS	-	Router Solicitation message
SEND	-	Secure Neighbor Discovery
TA	-	Tentative Address
TCP	-	Transmission Control Protocol
TCP/IP	-	Transmission Control Protocol / Internet Protocol
UDP	-	User Datagram Protocol
VoIP	-	Voice over Internet Protocol
WLAN	-	Wireless Local Area Network

**LIST OF APPENDICES**

<b>APPENDIX</b>	<b>TITLE</b>	<b>PAGE</b>
<b>A</b>	mipv6.cc	88
<b>B</b>	mipv6-1.tcl	91
<b>C</b>	mipv6-30.tcl	94
<b>D</b>	mipv6-1.sh	97
<b>E</b>	mipv6-30.sh	98

## **CHAPTER 1**

### **INTRODUCTION**

#### **1.1 Introduction**

Computer networks have become a very important aspect in today's human world. Networking and Internetworking are now very well known terminologies among specialists in every field of knowledge. For instance, banks use computer networks to link their branches as well as to assist in doing businesses. In another scenario, libraries use networks to link and share their digital resources with the world. Nowadays Internet has become known to the entire world. The Internet is now been used to link countries, corporations, companies, banks, hospitals, customers and so on. The technologies used in the Internet are merely computer networking technologies. Since its first launch, the Internet is facing a rapid growth in user numbers as well as more sophisticated technologies are being implemented. Such advancements make the costs for users to connect to the Internet get lower each day. Services such as sending voice signals over Internet technology, which is known as Voice over Internet Protocol (VoIP) (Davidson et al., 2006), are cheaper compared to the Plain Old Telephone Service (POTS). Sending video signals over the Internet is

also cheaper and more convenient for meetings and conferences compared to the traditional methods where one should travel for such meetings.

The Internet uses TCP/IP protocol suite for communication. Current Internet uses version 4 of the TCP/IP protocol suite which is also known as IPv4. The rapid growth of users and higher sophisticated services are forcing the Internet community to migrate to a more convenient TCP/IP protocol suite (IPv6) (Forouzan, 2006).

## **1.2 Networking Technologies**

### **1.2.1 IPv4**

IPv4 (Postel, 1981a) has been the most dominant protocol to be used and support connectivity for the Internet. Throughout the years IPv4 has seen many updates and changes to support the growth of the Internet. Technologies like Network Address Translation (NAT) (Egevang, 1994) have been introduced to support the reuse of IP addresses within private networks and corporations, which connect to the Internet through an Internet Server. Mobile IP has been introduced in order to support mobility and connectivity while roaming through different wireless networks routers without the need to reconfigure the mobile host (Perkins, 2002).

### **1.2.2 IPv6**

IPv6 was first introduced to solve the problem of depleting IPv4 addresses, where IPv6 uses an astonishing 128-bits addressing space compared to just 32-bits used in IPv4. However, technologies like NAT have slowed down the real migration to IPv6. IPv6 also addresses some other problems currently faced in IPv4, For example, it provides a better support for multimedia sharing and better mobility support (Deering and Hinden, 1998; Johnson, Perkins and Arkko, 2004).

### **1.2.3 Wireless Technology**

Wireless networking uses the technologies that help sending signals over transmission mediums without the needs of physical infrastructure. These technologies allows for a fair data rate to be sent and received through the wireless link. Wireless networking can be a better solution in situations where cabling is an option. Wireless LANs offer a combination of data connectivity with freedom of mobility (Wheat et al., 2001).

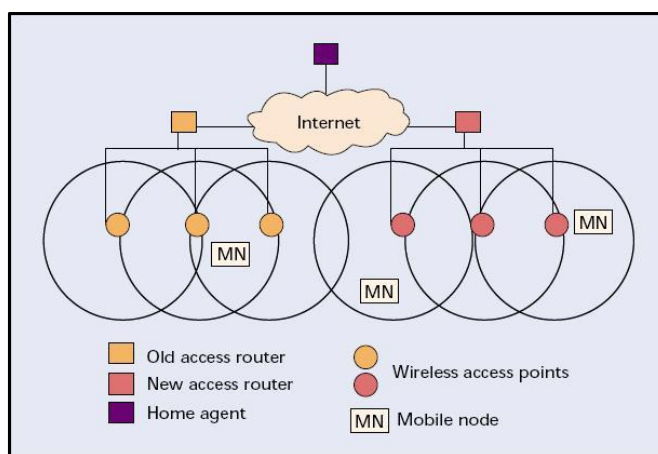
Wireless networks offer flexibility and reduce the cost and time needed to build up and break down a temporary network or groups of networks. This can be a big advantage for companies which hold up meetings and conferences frequently, where visitors can connect to the network using their own laptops.

Wireless networks also can be proven to be very convenient for old buildings where the costs of installing a WLAN can be very low compared to the costs of installing a wired LAN.

### 1.3 Problem Background

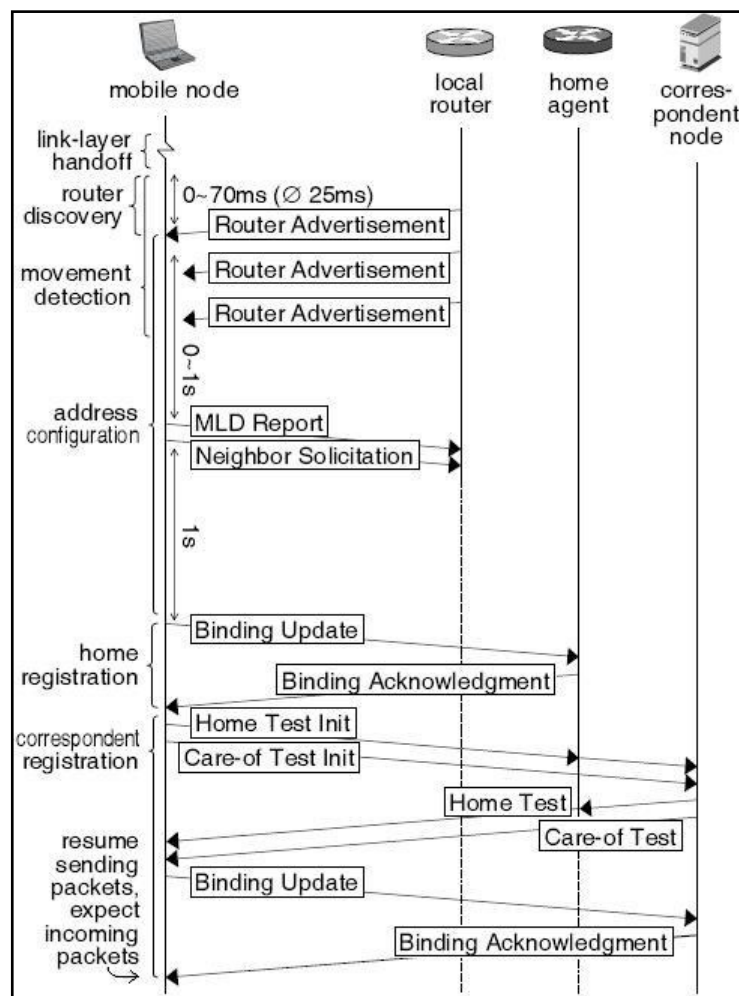
Since its first launch, the Internet faces an increasing number of challenges every day. Because of the rapid growth in the numbers of Internet users, the use of higher bandwidth and the introduction of more sophisticated technologies the cost to use the Internet these days is getting lower and lower. Handheld devices have become very small and capable of connecting to the Internet. Mobile IP (Postel, 1981a) was introduced to serve the demand of going mobile while being connected at the same time. It allows a moving node to stay connected while roaming across different sub-networks. The wireless network access offers some interesting advantages such as; it allows movements during communication and at a fair transfer rate (Montavont and Noël, 2002).

IPv6 (Deering and Hinden, 1998) was proposed to solve the problems currently faced in IPv4. One of the problems is the incapability of the IPv4 to support communications for huge numbers of mobile nodes. Figure 1.1, illustrates the wireless architecture in IPv6 (Montavont and Noël, 2002).



**Figure 1.1:** IPv6 Wireless Network Architecture (Montavont and Noël, 2002).

Mobile IPv6 (Johnson, Perkins and Arkko, 2004) was designed to manage mobile nodes movements across wireless IPv6 networks. However, during the movements, the active communication will be interrupted due to the changes of the network prefix of the new network. During this interruption, the mobile node cannot receive any IP packets on its new point of attachment until the handoff procedure ends. This interruption can take up to two seconds during a standard handoff procedure in MIPv6 (Vogt, 2006). This delay includes the time needed for new router discovery, movement detection, a new care-of-address establishment, and notifying the original router (Home Agent) and corresponding node of the new network attachment. Figure 1.2, shows the steps involve in the handoff procedure (Vogt, 2006).



**Figure 1.2:** Standard MIPv6 Handoff Procedure (Vogt, 2006).

Route Optimization (Johnson, Perkins and Arkko, 2004; Vogt and Doll, 2006) was introduced in Mobile IPv6 (Johnson, Perkins and Arkko, 2004) to solve the problem of triangular routing (Forouzan, 2006) as faced in MIPv4, by creating a direct path between the two communicating nodes. It allows an end-to-end way of communication between peers. Such situation is illustrated in Figure 1.2.

#### **1.4 Problem Statement**

In this project, several existing Route Optimization handoff techniques will be analyzed and simulated to study the efficiency of each technique. The standard Route Optimization approach in Mobile IPv6 shows a delay of up to 2 seconds before resume communication between two peers (Vogt, 2006). The performance of the Real-time applications streaming will be the focus of the study. In Real-time streaming (e.g. VoIP or video conferencing), the delay during the handoff can become a huge setback in implementing MIPv6 (Johnson, Perkins and Arkko, 2004).

A new approach will be proposed to reduce the time gap between link-layer handoff and resuming routability. A simulation then will be carried out to examine the effectiveness of the new approach against the existing schemes.

## **1.5 Project Objectives**

With reference to the problem stated above, this study is attempting to achieve the following objectives:

1. To study the existing handoff schemes of Route Optimization in Mobile IPv6.
2. To propose an alternative approach to reduce the latencies and packets loss for real-time communication streaming caused by the current Route Optimization techniques.
3. To analyze the performance of the proposed technique.

## **1.6 Project Scope**

1. The study of the existing handoff schemes of Route Optimization in MIPv6 will focus upon five of the most popular techniques.
2. The performance analysis of the proposed technique will be carried out by using ns-2 simulation tools.
3. TCP and UDP Packets of real-time streaming communication will be used as comparison metrics in the performance analysis between the proposed technique and the existing ones.
4. Security issues during handoff will not be addressed. This study will consider that there are neither security issues nor any kind of authentication or network keys for the Access Routers during the handoffs.
5. The simulation topology and model described in chapter 3 will not be changed for any of the handoff schemes.

## **1.7 Importance of Study**

This study gives an insight on the issues of handoff latencies in MIPv6. By studying and comparing several existing techniques proposed to solve the handoff latencies and optimize Route Optimization in MIPv6, this projects aims to propose a new approach to solve the issues regarding real-time multimedia streaming in order to improve the performance of the Transmission Control Protocol (TCP) and User Datagram Protocol (UDP) during the handoffs.

## **1.8 Organization of the Report**

The report is organized as follows:

Chapter (1) introduces the general idea of wireless network and the protocols that help achieve node mobility. This chapter also briefly discussed the Route Optimization handoff procedure and its issues faced in mobile IPv6. The project objectives and scope are mentioned in this chapter.

Chapter (2) discusses the literature reviews, where the background details and concepts of Transport Control Protocol (TCP), User Datagram Protocol (UDP) and Mobile IP were explained. The chapter also explains the Mobile IPv6 handoff procedure. The chapter also discusses some existing related works that concern with improving the performance of handoff and Mobile IPv6. The end of chapter two discusses some of the handoff schemes which will be simulated. A brief comparison in the chapter illustrates the advantages and limitations of each scheme.

Chapter (3) discusses the methodology used in this study. In this chapter the operational framework is presented to show the steps that are needed to be followed to achieve the objectives of this project. The Network Model will also be described in this chapter. The Network Model is designed as simple as possible to avoid unnecessary drawbacks, whether they are caused by the complexity of the design or the infrastructure of the network model. The Simulation Model that shows the parameters and metrics of the simulation is also described in this chapter.

Chapter (4) discusses the proposed scheme along with the algorithm and coding of the scheme. The simulation coding will be illustrated here as well.

Chapter (5) discusses the simulation model and the simulation metrics that are going to be studied for each chosen scheme. These metrics represents the times needed to be reduced in order for the scheme to perform better. The chapter also illustrates the results of the conducted simulations. Different existing MIPv6 handoff schemes along with the proposed alternative are simulated using the same network model. A comparison is carried out to compare the performance of each scheme.

Chapter (6) discusses the findings of the simulations and the study. The achievements of the study are discussed in this chapter as well. Some future works is proposed that can tune the proposed scheme and help implementing it in real MIPv6 networks.

## **1.9 Summary**

The ease of use, low cost and the vast variety of information made the Internet a necessity in the modern world. Users want to access information, their emails or even make businesses while they drive their cars or commute to work. Mobile IP was introduced to solve the problem of changing network attachments while keep communication available. The issues of handoff procedures (the process of changing network prefixes) delay the full implementation of Mobile IP over mobile networks. This report discusses some of these issues and proposes an alternative approach to help reduce the handoff latency of Mobile IPv6.

## **CHAPTER 2**

### **LITERATURE REVIEW**

#### **2.1 Introduction**

When TCP/IP suite was first introduced, routing was focused on fixed networks. That is when a node is configured with an IP address, which will be maintained throughout the communication session. Mobile IP was introduced to solve the problem of losing connectivity and open sessions between two communicating nodes, in situations where at least one of them is changing networks. Due to the roaming feature of mobile nodes, the IP address of that node needs to be changed accordingly, thus the packets coming from the corresponding node will never reach their destination unless the network infrastructure supports mobility.

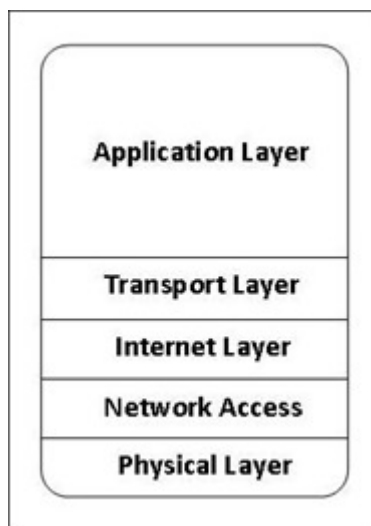
IPv6 has addressed some issues regarding sending and receiving multimedia streaming that were faced in IPv4. On the other hand, Mobile IPv6 is designed to maintain all the features of IPv6 as well as to manage mobile nodes' movements across wireless IPv6 networks.

This chapter discusses the TCP and Mobile IPv6 concepts. It begins with an overview of Transmission Control Protocol (TCP) User Datagram Protocol (UDP), Mobile IPv6 with Route Optimization and discusses the problems and current solutions to solve the Mobile IPv6 handoff procedure problems.

## **2.2 The TCP / IP Protocol Suite**

The TCP/IP protocol is a very well known protocol suite. It was designed to allow different computers from all sizes and vendors and running different operating systems, to communicate with each other. In other words it was designed independently from any hardware or software design. The TCP/IP protocol suite was named after its two most famous protocols Transmission Control Protocol abbreviated (TCP) and Internet Protocol (IP). The TCP/IP protocol stack, however, contains other protocols like (User Datagram Protocol (UDP), Internet Control Message Protocol (ICMP), and Domain Name Service (DNS)) (Forouzan, 2006).

Unlike the Open System Interconnection (OSI) model (Forouzan, 2006), the TCP/IP protocol is divided into five layers as shown in Figure 2.1, below. The physical layer deals with the physical interface (NIC) and the transmission medium. The network access layer deals with network access in which the IP is runs such as Ethernet, token ring, FDDI, etc. The network layer also routes packets across a network. A link layer is separated out at the end of the network layer that deals with the framing and physical access of the communication medium (Naugle, 1998). The specifications of this layer are not specified as part of the TCP/IP protocol suite. Instead, the details are defined in relevant link layer specification (Fikouras et al., 1998).



**Figure 2.1:** TCP/IP Suite Protocol Layers.

The network layer protocol, also known as the Internet Protocol, can be considered as the main thread that holds the entire Internet together. It is responsible for transferring data between hosts and nodes using various routing algorithms.

Layers that are above the Internet layer take a data stream and break it into chunks of a predetermined size called packets or datagrams. These datagrams are then sequentially passed to the Internet layer, which in turn route these chunks to the desired destination. Before transmitting data, the network layer might subdivide or fragment it into smaller packets for ease of transmission. When all pieces finally reach the destination, they are reassembled by the Internet layer into the original datagram. The IP protocol provides a connectionless, unreliable, best effort packet delivery service (Forouzan, 2006; Naugle, 1998). The IP header implements two basic functions, addressing and fragmentation. The IP header format is described in details in RFC 791 (Postel, 1981a).

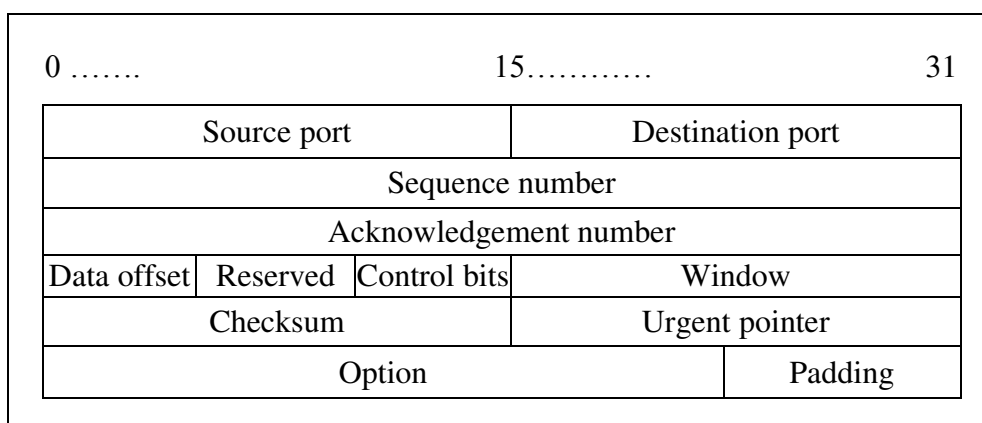
The Transport layer provides reliable, transparent transfer of data between end points. It also provides end-to-end recovery and flow control (Forouzan, 2006; Naugle, 1998). There are two different transport protocols in TCP/IP protocol suite, the Transmission Control Protocol (TCP) and User Datagram Protocol (UDP).

The most upper layer is the application layer which handles the details of a particular application. There are many protocols defined in this layer for different applications such as (FTP, TELNET, HTTP, etc) (Forouzan, 2006; Nagle, 1998).

### 2.2.1 The TCP Protocol Concept

The Transmission Control Protocol (TCP) resides in the transport layer of the TCP/IP protocol suite. While IP is an unreliable protocol, TCP is designed to provide reliable communication session between two pairs of processes (applications on different hosts) across a variety of reliable and unreliable networks and across the Internet (Forouzan, 2006). The TCP protocol provides a connection-oriented, end-to-end and reliable byte stream service between peers.

The TCP data unit is called a TCP segment. The TCP segment header as shown in Figure 2.2 is rather large, with a minimum length of 20 octets. The header include the source and destination port numbers, segment sequence number, acknowledgement number, checksum and some other fields (Forouzan, 2006; Nagle, 1998).



**Figure 2.2:** The TCP Packet Header.

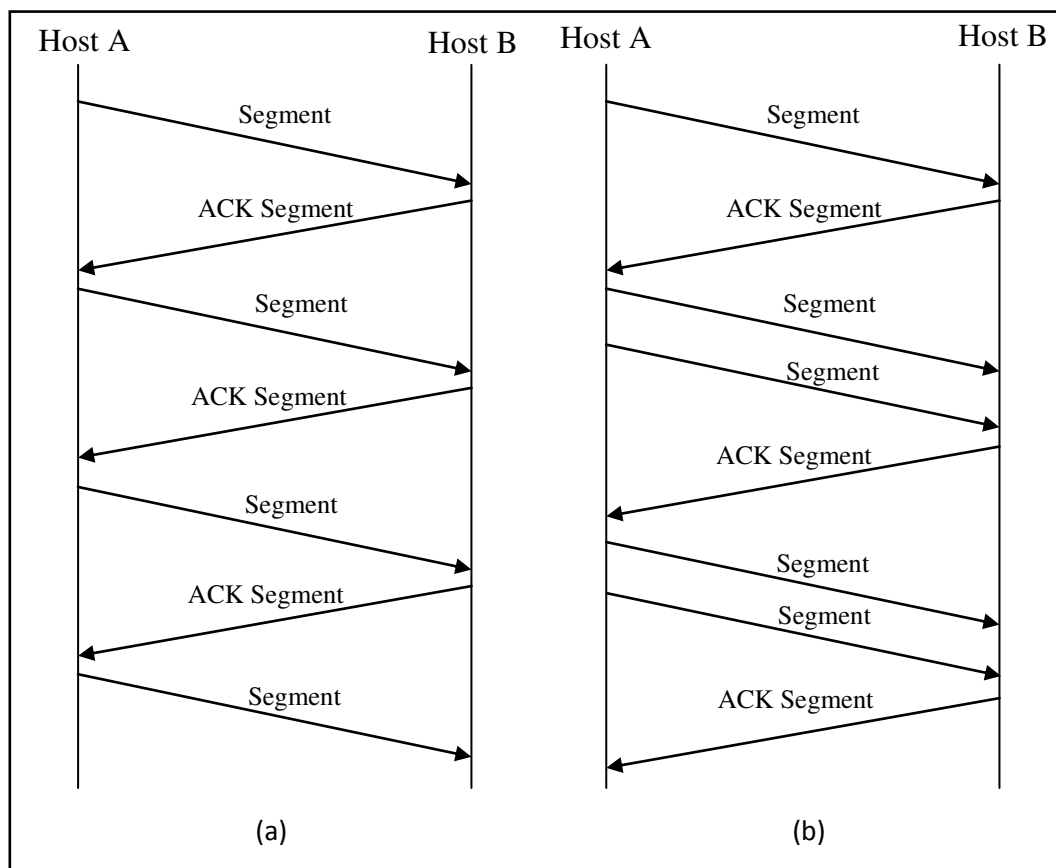
The first 16-bits of the TCP segment header consists of the source port number followed by 16-bits destination port number. To avoid duplicates, lost or out of order delivery during transmission, the header contains a 32-bits sequence number. The segment that has been successfully received by the destination host will be acknowledged with 32-bits of data. A 4-bits data offset number indicates where the data begins. The other details of the TCP segment header can be found in the RFC 793 Transmission Control Protocol Specifications (Postel, 1981b).

There are other attributes in the TCP Protocol that work with the application layer to help provide reliable services such as error detection, error correction, sliding windows, flow control, congestion control and the important slow start and fast retransmission (Forouzan, 2006; Postel, 1981b). The TCP protocol detects the loss of data, duplication or out-of-order delivery by assigning a sequence number to each octet transmitted and require acknowledgement from the receiving node. The rate of data flow through the TCP connected session is determined by the window sizing and retransmission strategy of TCP. A bigger window allowed by the receiver will result in a higher throughput (Stevens, 1997).

Acknowledgments in current TCP protocol are sent either immediate and for each segment or cumulative, that is acknowledging a number of received segments together (Clark, 1982). The immediate acknowledgement policy expects an acknowledgement message to be transmitted after each successfully received segment. On the other hand, in the cumulative policy, segments received will be recorded while waiting for an outbound segment to be acknowledged. It means that when a sender receives the acknowledgement, all previous segments have been received successfully. Thus, avoid long delays. A window timer can be set to allow the receiver to send an acknowledgement if the timer expires (Clark, 1982).

The cumulative acknowledgement policy is considered better in terms of efficiency. That is due to the fact that it produces less traffic load than the immediate acknowledgement policy and also reduces the TCP congestion window size.

Figure 2.3 shows both acknowledgement policies. It clearly shows that more segments are sent using the cumulative acknowledgement policy compared to the immediate acknowledgement policy over almost the same period of time.



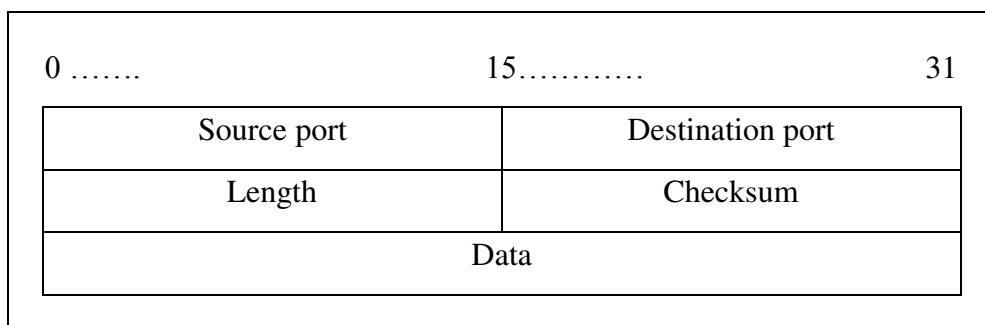
**Figure 2.3:** Acknowledgment Policies (a) Immediate (b) Cumulative.

### 2.2.2 The UDP Protocol Concept

The User Datagram Protocol is a connectionless transport layer protocol in the OSI model, which provides a simple and unreliable message service for transaction-oriented services. UDP is basically an interface between IP and upper-layer processes. The protocol ports distinguish multiple applications running on a

single device from one another. Unlike the TCP, UDP adds no reliability, flow-control, or error-recovery functions to IP and for that they require no acknowledgement replays. Because of UDP's simplicity, UDP headers contain fewer bytes and consume less network overhead than TCP.

UDP is useful in situations where the reliability mechanisms of TCP are not necessary, such as in cases where a higher-layer protocol might provide error and flow control, or real time data transportation is required. UDP is used as the transport protocol by several application-layer protocols as Network File System (NFS), Domain Name System (DNS) and Real-Time Protocol (RTP). Figure 2.4 describes the UDP segment.



**Figure 2.4:** The UDP Packet.

Since the UDP protocol does not have any recovery or error detection mechanisms, its header is rather small. A 16-bits source port is an optional field. When used, it indicates the port of the sending process and may be assumed to be the port to which a reply should be addressed in the absence of any other information. If not used, a value of zero is inserted. Another 16-bit is reserved to the destination port, which is used when replying a message. The Length indicates the length in octets of the user datagram, including its header and the data. The minimum value of the length is eight. Checksum is the sum of a pseudo header of information from the IP header, the UDP header and the data, padded with zero octets at the end, if

necessary, to make a multiple of two octets. The Data part contains the upper-level data information (Postel, 1980).

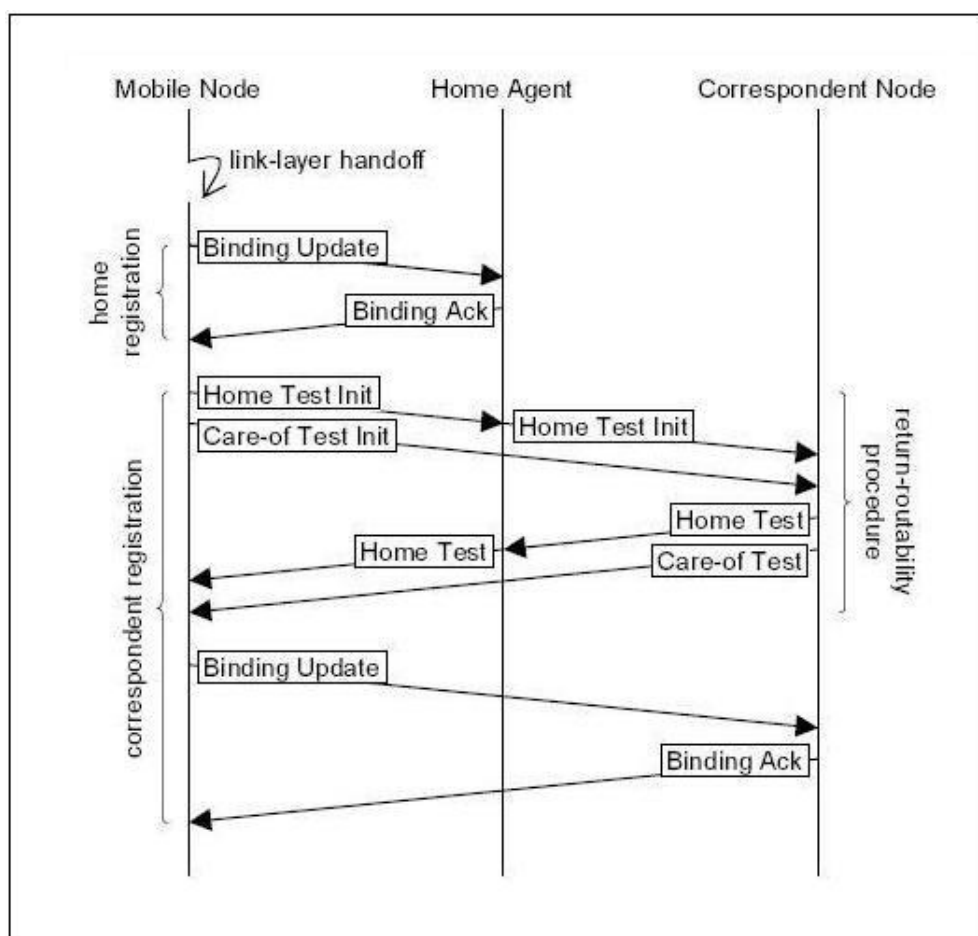
### **2.3 Mobile IPv6**

An IP address is used not just to pinpoint a specific host but also to tell the receiving router which interface a packet should be forwarded to. IP address contains a network ID which is used by the router to route the incoming and outgoing packets. This Network ID or attach link is fixed throughout the same network. The other portion of the IP address is the Host ID which represents specific host in the network. When two hosts communicate with each other they keep track of both their IP addresses and port numbers of each open session, which are used by the application layer (Forouzan, 2006). When a mobile host moves to a new network, its IP address will change according to the new network. This will cause the open session between two communicating nodes to be closed and discarded (Perez-Costa et al., 2003a).

Mobile IPv6 solves the ambiguity problem of IP addresses by using two of them per mobile node. Packets are routed based on a temporary care-of-address (CoA), which the mobile node replaces when it moves to a new network. A static home address, with a prefix from the mobile node's home network, serves as an identifier at upper layers (Johnson, Perkins, and Arkko, 2004). When it moves to a different network, the mobile node requests a dedicated router in the home network, its home agent. This home registration establishes a bidirectional tunnel between the home address and current care-of-address so that the mobile node can continue to communicate through the home address remotely. Mapping a care-of-address to a home address is called a binding. It is the mobile node's responsibility to update a binding whenever the care-of-address changes. When the mobile node receives the first encapsulated packet, it initiates a correspondent registration with the sender of

this packet. A route optimized packet carries the care-of-address within the IPv6 header during transit. The home address is located within an IPv6 Destination Options or Routing extension header, depending on whether the mobile node or the correspondent node sent the packet. Both end nodes swap the addresses when the packet traverses the IP layer so that transport protocols and applications can access the home address as usual (Johnson, Perkins, and Arkko, 2004; Vogt and Doll, 2006; Perez-Costa et al., 2003a).

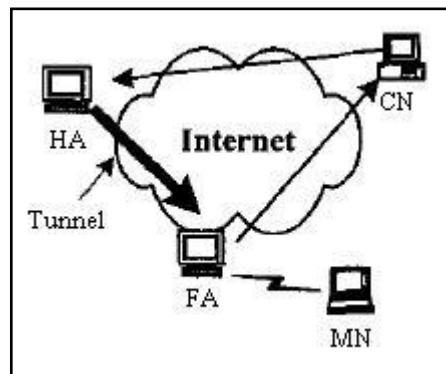
Figure 2.5 shows messages being exchanged during home and correspondent registrations. The home registration consists of a Binding Update message to notify the home agent of the new care-of address, and a Binding Acknowledgment message to indicate registration success or failure (Vogt and Doll, 2006).



**Figure 2.5:** Home and Corresponding Registrations (Vogt and Doll, 2006).

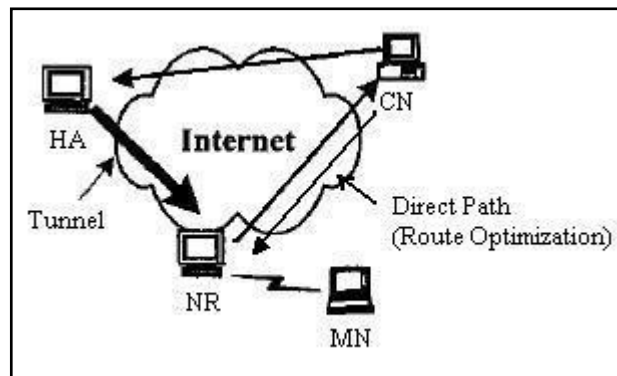
### 2.3.1 Route Optimization

With the development of new requirement to support multimedia streaming, the Internet Engineering Task Force (IETF) has added a new mechanism to support peer-to-peer communication. The performance of multimedia services, such as VoIP, is sensitive to long latencies caused by the tunneling process used in MIPv4 (Vogt, 2006). Mobility support in IPv4 requires the use of two router to serve as Home and Foreign Agents (HA) and (FA) respectively. These two routers manage all the packets coming to the Mobile Node (MN) by tunneling the coming packets from the HA to the FA which then sends them to the MN (Liu, Ye and Zhang, 2003). Figure 2.6 shows the tunneling process in MIPv4.



**Figure 2.6:** Tunneling in MIPv4 (Liu, Ye and Zhang, 2003).

Route Optimization (RO) allows peers to communicate directly without the use of Agents through a direct path. It is used to minimize the propagation latencies and packet overhead. However this mechanism introduced new latencies during handoff. The standard (RO) handoff procedure can take more than two seconds from start to finish depending on certain circumstances (Vogt and Doll, 2006). It introduces handoff delays, at IP layer, up to four round-trip times, depending on the implementation (Montavont and Noël, 2002). Figure 2.7 shows the direct path concept in MIPv6.



**Figure 2.7:** Route Optimization in MIPv6 (Liu, Ye and Zhang, 2003).

## 2.4 Handoff Procedure in MIPv6

Handoff occurs when a mobile node changes IP-layer connectivity. With a change in link-layer attachment or simply change the sub-network it connects to, the handoff takes place. The standard MIPv6 handoff procedure start with router discovery followed by address configuration, movement detection, and MIPv6 registrations (Vogt, 2006).

A MN detects that it has moved to a new subnet by analyzing the Router Advertisement (RA) periodically sent by the access-point router (AR). The MN can also request the AR to send a router advertisement by sending a router solicitation. The information contained in the router advertisement will allow the MN to create a new care-of-address (CoA) (Montavont and Noël, 2002). The MN first needs to verify the uniqueness of its link-local address on the new link (Deering and Hinden, 1998). The MN performs Duplication Address Detection (DAD) on its link-local address. Then, it may use either stateless (Thomson and Narten, 1998) or stateful (Droms et al., 2002) address auto-configuration to form its new CoA. Once it has obtained a new CoA, it performs a DAD to check if the address was already used by

another node (Thomson and Narten, 1998). However, DAD takes quite a long time with respect to the handover latency. In order to perform DAD, the MN has to send one or several neighbor solicitations to its new address and wait for a response for at least one second. This implies important additional time to handover latency.

Once the new care-of-address construction is done, the MN must update the binding cache in its home agent and correspondent by sending a binding update message. The MN can request an acknowledgment by setting a specific bit in its message (Montavont and Noël, 2002). Figure 1.2 (in Chapter 1) illustrates these processes.

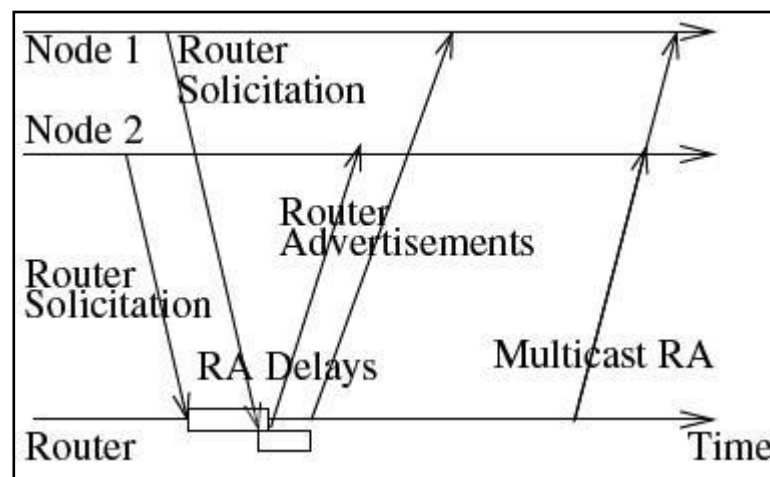
### **2.4.1 Router Discovery**

A mobile node learns about local routers and on-link prefixes during router discovery. This process is facilitated through Router Advertisement (RA) messages, which can be either a result of periodic broadcast or in response to a solicitation. The IPv6 Neighbor Discovery (RFC2461) (Narten, Nordmark and Simpson, 1998) states that unsolicited Router Advertisement messages are to be sent in random intervals of between 3 and 4 seconds at least and between 1350 and 1800 seconds at most. It also prevents a router from sending immediate responses to Router Solicitations (RS). This prevents mobility signaling for up to 500ms, which is the second highest delay just after the Duplicate Address Detection. Since these conservative limits are tailored towards stationary nodes and fail to meaningfully support mobility, the Mobile IPv6 (RFC3775) (Johnson, Perkins, and Arkko, 2004) decreases the lower bound to one beacon every 30 to 70 milliseconds. This reduces the mean time between successive advertisements to 50 milliseconds so that a mobile node can expect to receive the first post-handoff advertisement after 25 milliseconds. On the other hand, high frequencies for multicast advertisements may be an issue in low-

bandwidth, wide-area networks, where many users may not frequently leave the geographic area covered by the same IP subnet (Vogt, 2006).

All these strict bounds are made on the rate at which Multicast Router Advertisements may be sent. However, Unicast RAs are not limited by these bounds, as they respond to Router Solicitation Messages, which make them suitable for fast router discovery. Unicast RAs can only be sent in response to solicitations which come from a valid unicast source address. Therefore these solicitations may only be sent after the soliciting host's link local address is configured.

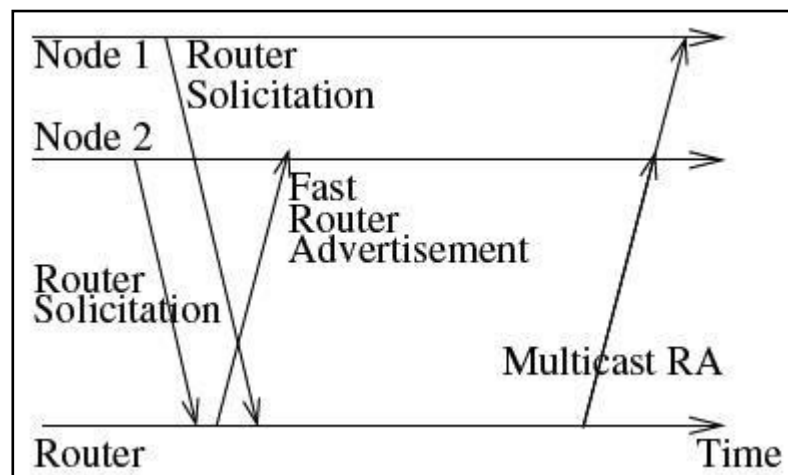
The Neighbor Discovery RFC (Narten, Nordmark and Simpson, 1998) specifies a random timeout between 0 and 500 milliseconds before a router transmits a unicast RA in response to a RS. This is principally to prevent multiple routers on a link from simultaneously transmitting to soliciting nodes. It does not rate limit any unicast responses. This operation is illustrated in Figure 2.8 (Narten, Nordmark and Simpson, 1998). The delays of unicast responses affect the overall performance of handoff in MIPv6 (Daley, Pentland and Nelson, 2006).



**Figure 2.8:** Unicast and Multicast Router Advertisement Messages in Standard Mode (Daley, Pentland and Nelson, 2006).

### 2.4.1.1 FastRA

Fast Router Advertisement (FastRA) (Kempf, Khalil and Pentland, 2005) allows for at most one router on a link to be configured to provide immediate unicast responses to Router Solicitations. Using a single router as the fast respondent ensures that no collisions occur. Selection of the fast respondent is assumed to be by manual configuration although some additional automatic protocol could be specified. FastRA contains a rate-limiting mechanism based on a counter which allows several unicast responses to be processed before successive solicitations are dropped, pending a multicast RA. This value is known as MAX\_FAST\_RA. The Fast RA counter is reset after a multicast RA and unicast FastRA responses are similarly re-allowed at this time. Figure 2.9 shows the operation with MAX\_FAST\_RA = 1 (Daley, Pentland and Nelson, 2006). On a correctly configured network, MAX\_FAST\_RA should be set high enough to handle all expected node configurations, up to that safely possible on the link (Kempf, Khalil and Pentland, 2005). This will guarantee that the maximum number of MN's will receive fast advertisements.



**Figure 2.9:** Unicast and Multicast Router Advertisement Messages in FastRA Mode (Daley, Pentland and Nelson, 2006).

In (Daley, Pentland and Nelson, 2006), the authors tested the performance of FastRA compared to RFC 2461 RA (Narten, Nordmark and Simpson, 1998). They used an environment free of DAD, where a mobile node can roam across access points and use its previous IP address every time it returns to its home network. During the test for the FastRA case, all handovers were under 50 milliseconds. It was also clear that FastRA has been effective in removing the principle cause of Layer 3 handover delay, which is delay between sending of an RS and receiving an RA response. However, issues relating to address acquisition may prevent its use where unoptimized duplicate address detection is required.

#### **2.4.1.2 RA Caching**

In (Choi and Shin, 2002) the authors introduced fast router discovering with RA Caching. No trigger information is needed on the MN side, although the access network sends a Router Advertisement as soon as the MN joins the network. Since the router does not necessarily have access to information about the MN's arrival, the task is delegated to a wireless AP which receives link association messages. This AP, which has been caching unsolicited Router Advertisements, forwards the most recent one to the MN, so that it can learn quickly if the IP network has changed. Whilst the caching of network-layer messages in a link-layer device (the access point) has been seen as a conflict between OSI layering of communications tasks (this is true of triggering RSs as well), the processes associated with delivery of the RA may be achieved without modification of the IP datagram, and no more network layer function than is available in a modem packet capture program. It is notable that a similar mechanism has been proposed for Mobile IPv4 (Daley, Pentland and Nelson, 2003).

Both RA Caching and FastRA require modifications to the access network infrastructure, but neither requires explicit support on the MN other than simple link-up triggers already supported in MIPv6.

### **2.4.2 Movement Detection**

Movement detection is the second process happens after router discovery in handoff procedure. A MN uses movement detection to recognize changes in the IP connectivity. The change implies that the MN may choose a new default router, reconfigure a new unique link-local address and initiate home and CN registration. Actually there is no specific movement detection technique for a MN to use to assure its movement to a new subnetwork.

Movement detection relies on Router Discovery improvement technique to achieve better and faster decisions. It relies on analyzing the on-link prefixes advertised in Router Advertisement messages and possibly also probing reachability of routers considered off-link. When the prefixes that are used by the MN are no longer seen to be advertised, but new prefixes show up instead, the mobile node typically decides that it has moved to a different network. On the other hand, the received prefixes may also indicate that IP connectivity did not change in spite of a link-layer handoff, (e.g., when the mobile node switches access points that connect to the same subnet). Reception of a single advertisement is usually insufficient to decide whether IP connectivity has changed or not. It is also generally impossible to determine when an advertisement should have been received, but did not appear, due to the lack of a guaranteed advertisement interval (Vogt, 2006). Nevertheless, the absence of a single expected advertisement still does not imply a change in IP connectivity given the potential for packet loss. Three missing advertisements indicate movement more reliably. A decision can then be made at most 270

milliseconds after the last advertisement was received from the old default router. The actual link-layer handoff may occur up to 70 milliseconds later, so movement detection can take any time between 200 and 270 milliseconds. On average, the period between reception of the last advertisement from the old default router and the link-layer handoff is 25 milliseconds, yielding a mean movement-detection delay of 245 milliseconds (Narten and Draves, 2001).

The movement detection optimizations are able to consistently achieve better movement detection and handover times than unoptimized systems, and in most cases can achieve handover durations which are better than the time taken to detect movement with RA beacons. The handover duration using the optimization techniques falls within the acceptable range for existing VoIP loss concealment techniques (Daley, Pentland and Nelson, 2003).

### **2.4.3 Address Configuration**

A Mobile Node needs an effective IP address to be able to communicate with other nodes. When a MN decides that it has moved to a new network, after receiving a RA with a new prefix, it initiates an address configuration process. In standard MIPv6 (Johnson, Perkins, and Arkko, 2004), this process requires the use of Stateless Address Auto-configuration (Kempf, Khalil and Pentland, 2005). The mobile node chooses an interface identifier, either randomly or based on the interface's MAC address, then combines the network prefix obtained from the router with a suffix generated from its 64-bit Interface Identifier. This untested address is called Tentative Address (TA). It then sends a Multicast Listener Report message to subscribe to the solicited-node multicast group corresponding to the new address. If the RA message was a multicast transmission, which usually is the case, the Multicast Listener Report message is delayed by up to one second to desynchronize

with neighboring nodes that may be reacting to the same advertisement. Duplicate Address Detection protocol must be run to verify whether the new TA is unique. The MN transmits a Neighbor Solicitation message for the address and, if no responses are received within a period of one second, assigns the TA to the interface (Thomson and Narten, 1998).

The total configuration period hence, ranges between one and two seconds if the address is unique. Even though the link-local address keeps its prefix during handoff, the mobile node must still re-verify uniqueness of this address when IP connectivity changes, because a node on the new link may already be using the same link-local address. This is done through another run of Duplicate Address Detection. Since only movement detection can establish whether IP connectivity has changed, re-verification of the link-local address typically begins after movement detection (Vogt, 2006).

#### **2.4.3.1 Duplicate Address Detection**

DAD uses stateless address strategy which attempts to detect duplicate addresses without relying on a centralized router to keep track of the state of the network. Instead, DAD relies on the already configured nodes to cooperate in the DAD process (Thomson and Narten, 1998). The disadvantage is that this strategy depends on nodes to defend their addresses, and there is no positive acknowledgement for a node to allow it to use the address. If a solicitation or a defense packet is lost a collision can go undetected. In addition, a configuring node must wait for a set time to allow negative messages to be received rather than receiving a positive message and continuing immediately (Narten and Draves, 2001).

### 2.4.3.2 Alternative Address Configuration Methods

Dynamic Host Configuration Protocol for IPv6 (DHCPv6) (Droms et al., 2002) may seem to solve all the problems of DAD by offering a stateful, server-based method of negotiating addresses. However, communication with a DHCPv6 server requires that the Mobile Node has already configured a Link-Local address, which requires the use of standard DAD and thus introduces the one second delay. Also, standard DAD requires that the DHCP-assigned addresses to be double-checked with DAD (Thomson and Narten, 1998), as the DAD server will be unaware of nodes which have configured statelessly. This potentially increases DAD latency, as first the Link-Local, and then the Global addresses must clear DAD, rather than being able to do both in parallel. As a stateful method, DHCPv6 is dependent on server-stored state, reducing the reliability of the network to that of the DHCP server. For that reason, DHCPv6 is not considered an appropriate replacement for DAD (Moore, 2006).

#### A) Advance Duplicate Address Detection

In (Han et al., 2003), Advance Duplicate Address Detection (A-DAD) has been proposed. A-DAD capable routers supply addresses to arriving MN from a pool of addresses which are known to be unique on the link. A host can safely configure this address without performing DAD, as the router has ensured its uniqueness. In order to provide addresses for this pool, the router must create addresses based on random suffixes as per RFC 3041 (Narten and Draves, 2001) and undertake standard DAD on them. This means that the router must configure sufficient addresses in advance to ensure that demands are met. The configuring node has no choice as to what address is provided. Since Secure Neighbor Discovery (SEND) is likely to rely upon Cryptographically Generated Addresses (Aura, 2005), nodes which are depend

on A-DAD rather than generating addresses from their own private keys will be excluded from Secure Neighbor Discovery.

## **B) MLD-DAD**

In their paper (Daley and Nelson, 2003), the authors introduced a new address configuration alternative. MLD-DAD uses routers to monitor MLD reports for the solicited nodes addresses and determine which addresses are being used. As periodic updates are required by the MLD protocol, the router's state will be kept up-to-date, and reconstructed if the router is reset. If the router has seen no evidence of the address being used, it can inform the node that the address may be allocated without further delay. However, MLD-DAD may only be used on networks where all devices on the network perform MLD properly. Otherwise, the router cannot be sure of the availability of an address.

## **C) Optimistic DAD**

RFC 4429 (Optimistic DAD for IPv6), (Moore, 2006), suggests that an optimistic strategy, where the node assumes that DAD will succeed, would be preferable and speed up the address configuration process. Such an approach bends the rules of RFC2461 (Narten, Nordmark and Simpson, 1998) and RFC 2462 (Thomson and Narten, 1998) to allow communication to be established over a Tentative Address, while attempting to minimize disruption in case of collision. To avoid this problem, Optimistic DAD exploits existing flags and options in the ND

messages. NAs are sent with the (Override) bit cleared, while NSs and RSs are sent without Source Link-Layer Address Options. The Optimistic node modifies its ND behaviors to work around these restrictions. The restrictions prevent a Tentative Address overriding existing Neighbor Cache entries in the case of a collision, although it does make the ND process less efficient while the address is Tentative. Once the DAD timeout has expired, the address is no longer Tentative, and standard ND behavior applies.

In the case of an address collision, the Optimistic node is unlikely to be able to properly communicate, since its neighbors will not allow it to complete Neighbor Discovery. As soon as the defending NA is received it will reconfigure a new address in any case.

There is still a possibility that the collision will cause the connection to be lost, but the situation will be rapidly resolved, as opposed to the irresolvable problems caused by a collision without DAD. Optimistic DAD is most suitable for networks in which the transmission of a few extra messages per configuring node is not a significant issue. In addition, because the penalty associated with an address being Tentative is greatly reduced, a node may elect to probe more than once for a duplicate address, greatly decreasing the chance of packet loss causing a collision to go undetected (Narten and Draves, 2001). This makes Optimistic DAD particularly suitable for use on Wireless LAN type networks where packet loss is common (Willig, Kubisch and Wolisz, 2001).

#### 2.4.4 Home Agent and Corresponding Node Registrations

The MN uses its new Care-of-Address to register itself with its Home Agent (HA) and Correspondent Nodes (CN). This establishes a binding between the CoA and the MN's home address, which has a prefix from the HA's network and remains stable across movements. The Home Address is used at stack layers above IP as part of end-point identification. Data packets that a MN exchanges with a peer have the CoA in the IP header and the Home address in an IPv6 extension header while on the wire. Both end nodes swap the addresses when a packet traverses the IP layer so that transport protocols and applications can access the Home address as usual. The Home Registration consists of a Binding Update (BU) message which notifies the HA of the new CoA, and a Binding Acknowledgment (BA) message indicating success or failure.

The correspondent registration permits Route Optimization. It includes a BA message that conveys the new CoA to the CN, and a responding BA message. Reachability at both Home and CoA entitles the MN to initiate a binding between the addresses. For the Home address test, the Mobile Node tunnels a Home Test Init message to the Home Agent, which forwards the message to the Correspondent Node. The Correspondent Node returns an unpredictable home keygen token to the Home address within a Home Test message. The HA intercepts this message and tunnels it to the MN. The CoA test is a direct exchange between the MN and the CN. It consists of a Care-of Test Init message and a Care-of Test message with an unpredictable care-of keygen token. Knowledge of the home and care-of keygen tokens proves the MN's ability to receive packets at the Home address and CoA, respectively. The Mobile IPv6 RFC (Johnson, Perkins, and Arkko, 2004) leaves MN liberties with respect to scheduling signaling and data packets.

#### 2.4.4.1 Mobile IPv6 Optimizations

Many Mobile IPv6 optimizations reduce the handoff delays of Route Optimization through modifications of the return-routability procedure. A combination of Early Binding Updates (Han et al., 2003) and Credit-Based Authorization (Kempf, Khalil and Pentland, 2005) achieves this, on a purely end-to-end basis, with the following four constituent optimizations:

1. Proactive home-address tests:

A mobile node acquires a home keygen token for a future handoff during a proactive home-address test. This saves a possibly long round trip through the home agent during the critical handoff period. The mobile node can invoke proactive home-address tests on a just-in-time basis, if its link layer provides a trigger indicating imminent handoff, or periodically whenever the most recently obtained home keygen token is about to expire.

2. Concurrent care-of-address tests:

Data packets can already be exchanged, to a limited extent, via a new care-of address, while the mobile node's reachability at that care-of address is being verified.

3. Tentative bindings:

The mobile node registers a tentative binding between its home address and an unverified care-of address by exchanging Early Binding Update and Early Binding Acknowledgment messages with a correspondent node. The messages are authenticated only with the home keygen token obtained from a recent proactive home-address test, thus facilitating a subsequent, concurrent care-of-address test.

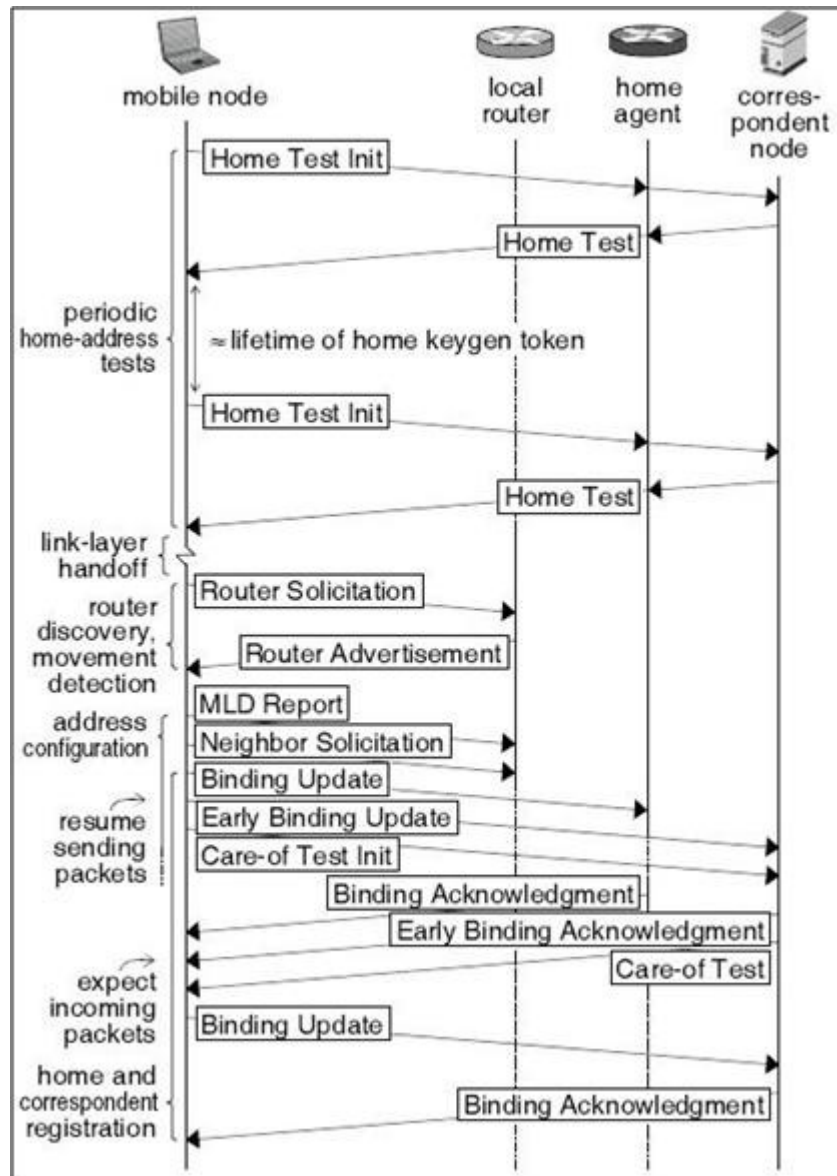
4. Parallel home and correspondent registrations:

The Mobile IPv6 specification does not permit the mobile node to send a Binding Update message to a correspondent node before it receives an acknowledgment from the home agent. This becomes a performance issue if

the combination of proactive home-address tests and concurrent care-of-address tests hides the latency of the return-routability procedure. The rules of Mobile IPv6 are hence relaxed so as to allow a mobile node to send an Early Binding Update message when the home registration is still pending.

In (Vogt, 2006) the author shows a solution to solve the problem of the handoff latencies by introducing a new approach to handle the registration process. It describes two kinds of MNs. First one is the Conservative MN and the second is an Optimistic MN. Figure 1.2 shows a conservative MN, which waits for the BA message from the HA before it initiates the return routability procedure. In contrast, an optimistic MN executes the home registration and the return routability procedure in parallel as in Figure 2.10. An optimistic MN furthermore starts sending packets to the CN as soon as the BU message for the CN has been brought on way, whereas a conservative MN uses the new CoA only after reception of an acknowledgment. In either case, the CN is unaware of the new CoA until it receives the BU message. Its first packet sent to the new CoA will hence be delivered to the MN roughly along with the BA message, assuming that one was requested by the MN.

Conservative MNs avoid a useless return routability procedure in case the home registration fails. They also do not risk loss of packets sent shortly after a lost or rejected BU message. This comes at the cost of additional handoff latency when both registrations are successful. For outgoing route-optimized packets, this is a round-trip time between the MN and the HA plus a round-trip time between the MN and the CN. For incoming packets, the additional handoff latency is a round-trip time between the MN and the HA.



**Figure 2.10:** Optimistic Mobile Node (Vogt, 2006).

The author in (Vogt, 2006) states that optimistic MNs perform better in the general case. But they may attempt a return routability procedure in vain or suffer packet loss should the home or correspondent registration fail.

## **2.5 Comparison between Existing Handoff Schemes**

Up until now researchers are still looking for an efficient handoff scheme that can be implemented with as less modification to the MIPv6 as possible. Some of the most popular ones are FastRA, RA Caching, Advanced DAD, Optimistic DAD and Optimistic Node. Table 2.1 shows the differences and techniques used between the schemes and their respective advantages and limitations.

### **2.5.1 FastRA**

One router is configured to provide immediate unicast responses to RS instead of waiting a random time between 0-500ms. A rate-limiting mechanism allows several unicast responses to be processed before successive solicitations are dropped (Kempf, Khalil and Pentland, 2005; Daley, Pentland and Nelson, 2006).

### **2.5.2 RA Caching**

Wireless APs cache unsolicited RA and forward the most recent one to new MNs. It sends one multicast RA message encapsulated in a unicast frame to MN and sends it only to cells with new MNs (Choi and Shin, 2002).

### **2.5.3 Advanced DAD**

A method for supplying addresses quickly to roving mobile nodes. An Advanced DAD capable router supplies addresses to arriving MN from a pool of addresses which are known to be unique on the link. A host can safely configure this address without performing DAD, as the router has ensured it is unique (Han et al., 2003).

### **2.5.4 Optimistic DAD**

This approach allows communication to be established over a tentative address, while attempting to minimize disruption in the case of collision. This permits the MN to resume communication without the delay of checking the address uniqueness (Moore, 2006).

### **2.5.5 Optimistic Mobile Node**

In this approach, the MN executes a proactive home-address test with the CN prior to handoff, sends an immediate RS and receives a RA, uses Optimistic DAD to configure new and re-verify existing addresses, registers with its Home Agent and CN a new CoA for which uniqueness verification is still in progress (Vogt, 2006).

**Table 2.1:** The Differences in Techniques used in Existing Handoff Schemes and their Respective Advantages and Limitations.

Scheme	Technique	Advantages	Limitations
FastRA	<ul style="list-style-type: none"> <li>- One router is configured to provide immediate unicast responses to Router Solicitations.</li> <li>- A rate-limiting mechanism allows several unicast responses to be processed before successive solicitations are dropped.</li> </ul>	<ul style="list-style-type: none"> <li>- Ensure an environment with fewer collisions.</li> <li>- Effective in removing the delays between sending RS and receiving RA response.</li> <li>- Improve movement detection.</li> </ul>	<ul style="list-style-type: none"> <li>- The router must be manually configured.</li> <li>- Nodes must have a configured address in order to send a RS.</li> <li>- Vulnerable for packet loss.</li> <li>- Packet loss cause for longer handoffs.</li> <li>- Faces RS bombing when many nodes simultaneously come up online.</li> <li>- Require modification to the access network infrastructure.</li> </ul>
RA Caching	<ul style="list-style-type: none"> <li>- Wireless AP cache unsolicited RA and forward the most recent one to new MNs.</li> <li>- Sends one multicast RA message encapsulated in a unicast frame to MN.</li> <li>- Sends RA only to cells with new MNs.</li> </ul>	<ul style="list-style-type: none"> <li>- No modification in the IP datagram is needed.</li> <li>- No additional network-layer functions are needed.</li> <li>- No need for RS to send RA.</li> <li>- Prevent bandwidth wastage when there are a large number of cells per subnet.</li> </ul>	<ul style="list-style-type: none"> <li>- Caching network-layer messages in link-layer device conflicts with OSI layering system.</li> <li>- Require modification to the access network infrastructure.</li> <li>- Must use RA caching capable AP.</li> <li>- APs should scan every L2 frame to look for unsolicited RA message.</li> </ul>
Advanced DAD	<ul style="list-style-type: none"> <li>- Routers create address based on random suffixes and undertake standard DAD on them.</li> </ul>	<ul style="list-style-type: none"> <li>- MN can configure the new CoA safely without performing DAD.</li> </ul>	<ul style="list-style-type: none"> <li>- Router must configure sufficient address in advance to meet any demands.</li> <li>- Configured MNs must use address provided by the router.</li> <li>- MNs are excluded from Secure Neighbor Discovery.</li> </ul>
Optimistic DAD	<ul style="list-style-type: none"> <li>- Exploits existing flags and options in the ND messages.</li> </ul>	<ul style="list-style-type: none"> <li>- Allow communication to be established over a Tentative Address.</li> </ul>	<ul style="list-style-type: none"> <li>- In case of collisions the MN will not communicate properly due to connection loss.</li> <li>- Not suitable for network with huge number of MNs.</li> <li>- Nodes rely on the default router to forward packets to their destinations.</li> </ul>
Optimistic Node	<ul style="list-style-type: none"> <li>- MN execute proactive Home-Address test with CN prior to handoff.</li> <li>- Use Optimistic DAD to configure new CoA.</li> <li>- Send Home and CN Binding Update in parallel.</li> <li>- CN sends data to MN via its new unverified CoA.</li> </ul>	<ul style="list-style-type: none"> <li>- Solicited RA are more rate economic than unsolicited RAs.</li> <li>- Combining fast handoff schemes together can reduce the handoff latency efficiently considering no errors occur.</li> <li>- Tentative Binding conceals latencies of return-routability procedure.</li> </ul>	<ul style="list-style-type: none"> <li>- Great possibility of packet loss if the address configuration fails.</li> <li>- Periodic Home-Address tests can demonstrate have loads especially in networks with many infrequently moving MNs.</li> <li>- Subject to packets out of order arrival.</li> </ul>

## 2.6 Simulation Tools

To achieve the objectives of this project; one main element is the simulation tool used to simulate the behavior of the network and the hosts within that network. Network Simulator 2 (ns2) (McCanne and Floyd), is a network simulation tool known among network researchers as an open source and easy to use tool for simulation experiments. (ns2) comes in different versions and extensions. One of the extensions is MobiWan (Ernest, 2002), which support the simulation of Mobile IPv6 over wide area LANs.

### 2.6.1 Network Simulator 2

(ns2) is an open source, freely distributed discrete event simulator for networking research. (ns2) provides substantial support for simulation of TCP, routing, and multicast protocols over wired and wireless networks. The ns2 covers a very large number of applications, protocols, network types, network elements and traffic models. They are all called simulated Objects (Altman and Jimenez, 2003; Heidemann and Huang, 2002; Fall and Varadhan, 2008).

In ns2 the advance of time depends on timing of events which are maintained by a scheduler. (ns2) is based on two languages: an object oriented simulator, written in C++, and an OTcl (Object oriented extension of (Tcl) interpreter, which is used to execute user's command scripts (Jung et al., 2003). The C++ gives complete control over packet-processing operations because it is fast to run and detailed. On the other hand, the easy to write and change OTcl, controls the simulation setup, configuration and occasional actions (Altman and Jimenez, 2003; Heidemann and Huang, 2002).

### 2.6.2 MobiWan: ns-2 Extension to Support mobility in IPv6 Networks

MobiWan is a simulation tool based on (ns2) meant to simulate Mobile IPv6 under large Wide-Area Networks (both local-area mobility and global-area mobility). MobiWan comprises extensions to simulate Mobile IPv6, and extensions to manipulate and configure large network topologies (TOPOMAN / TOPOGEN). Both extensions can work independently from each other (Ernest, 2001).

- TOPOMAN: library to create, configure, and manipulate large topologies in a very easy manner.
- TOPOGEN: translator from GT-ITM to NS-2 which output TOPOMAN procedure calls.

## 2.7 Summary

This chapter explains general concepts about the TCP/IP protocol. The TCP/IP is a crucial element in researching the performance of the handoff procedure and Route Optimization scheme. The Transmission Control Protocol (TCP) provides a connection-oriented communication between peers. Every data sent from or to a node is ensured to be delivered and to reach its destination. If the data is lost during the delivery, the TCP will insure its recovery by retransmitting the lost packets again. Some of the most popular ones are explained here with their advantages and implementation limitations. For that reason the same network simulation will be carried out to show the real performance of some of the schemes when compared to other schemes. Metrics such as handoff latency and packet loss are among the performance metrics used in this simulation.

The Mobile IPv6 is a protocol used to support nodes' mobility in the network. It allows nodes to move across different networks and Access Points without losing open communication sessions. Route Optimization (RO) is introduced and implemented in Mobile IPv6 to eliminate the need for a Foreign Agents (used in Mobile IPv4). It provides a direct communication path between peers, thus, reducing the time and distances that packets had to travel between Home and Foreign agents.

The handoff delays as a result of nodes mobility within a network affect the performance of the whole network. In other words, the use of the TCP is important in Mobile IP networks to ensure that the data sent is not lost during handoffs.

Network Simulator 2 (ns2) is considered as a suitable tool to simulate networks and nodes behaviors. The MobiWan extension is designed specially to support Mobile IPv6 in wide-area networks. These two tools together will be used to simulate the different RO handoff schemes in MIPv6.

## **CHAPTER 3**

### **METHODOLOGY AND FRAMEWORK**

#### **3.1 Introduction**

The goal of this project is to investigate the delay issues that happen during the handoff in MIPv6. The investigation requires a simulation environment where some of the existing schemes to reduce the handoff latencies will be tested. A new approach to reduce the handoff latencies will be proposed in later chapters and a simulation will be carried out to compare the new approach to the existing ones.

The simulation will be carried out using Network Simulator (ns2) (McCanne and Floyd). The network model for the simulation will be described in later sections. To make the simulation simple, the Corresponding Node (CN) will be connected to a wired LAN. One Mobile Node (MN) will be moving across Access Routers (ARs) representing different sub-networks.

## **3.2 Operational Framework**

The first part of this study is to carry out simulation experiments to compare different existing RO handoff schemes. The Network model is a simple and general design that can be found in different situations and environments. The experiments will see the use of different scenarios beginning with the simplest one and going up to a more complicated scenario. The Network Model will maintain the original design for the whole simulation time, while the number of communicating nodes will increase from time to time to study the effects of each handoff scheme.

The second part of this study is to design a new approach to reduce the handoff latencies faced in MIPv6 using Route Optimization. The proposed approach then will be simulated through a series of simulation experiments to analyze its performance.

The focus of this project is the performance of TCP when transmitting real-time streaming (e.g. VoIP or Video Conference). The real-time streaming is very sensitive to handoff latencies, packet loss and out of order delivery. The proposed scheme will include some features of the most promising schemes including Fast Router Advertisement (FastRA) (Kempf, Khalil and Pentland, 2005), and Router Advertisement caching (Choi and Shin, 2002). Figure 3.1 shows the Operational Framework of the study.

### **3.2.1 Phase One**

This is the analysis phase where the main problem is identified and studied. This phase includes two main parts: First, identifying and formulating the problem. This involves identifying the issues faced during the handoff procedure in MIPv6.

Second, the literature reviews in which the existing schemes are studied and analyzed. This includes identifying the limitation and weaknesses of the schemes and studying the possibility to improve in one or more areas. Some other features and problems of the standard Route Optimization handoff in MIPv6 are extracted at this stage to look for some room for improvement.

### **3.2.2 Phase Two**

This phase is divided to two parts where in the first part is to design the network model that will be simulated. The design is made as simple as possible to emulate a simple everyday environment that can be found around companies' buildings or university campuses.

The second part is to design the simulation model, where the parameters of the simulation will be set. The parameters of the simulation model will change according to the scheme that will be simulated as well as according to other elements, including the number of MNs within a range of an Access Router and how often each node experience a handoff procedure.

### **3.2.3 Phase Three**

In this phase some of the existing schemes will be chosen based on their performance and they will be simulated with the simulation model. The actual

performance of each simulated scheme will be analyzed to actually decide whether or not the scheme achieves the stated performance. At this stage the result of the different schemes will show exactly how each scheme performs against the other schemes within the same simulation model. The result of this phase will help decide the areas of probable improvement in each scheme and help in designing the new proposal.

#### **3.2.4 Phase Four**

This phase is where all the work from the previous phases will be put together to improve the handoff latency of MIPv6. The new approach will be designed according to the weaknesses and limitations of the existing schemes. The new design will be simulated in the second part of this phase to get the results of its performance.

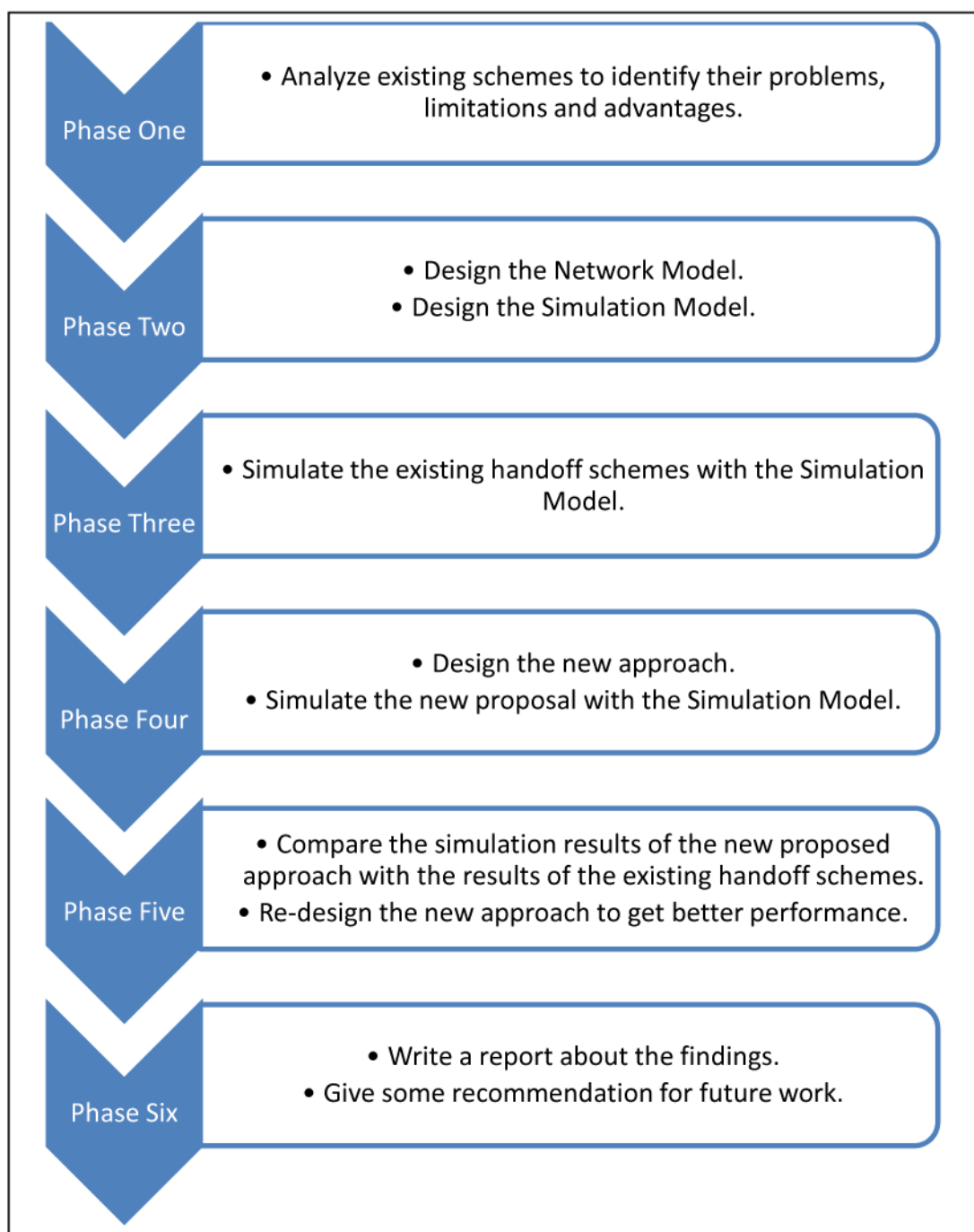
#### **3.2.5 Phase Five**

In this phase the results from the simulation of the new proposal will be compared to the results from the previous work. This is where the proposed scheme will be put to test against some of the promising schemes proposed to reduce the handoff latencies.

In the second part of this phase the design of the new approach will be studied and redesigned if necessary to achieve better results.

### 3.2.6 Phase Six

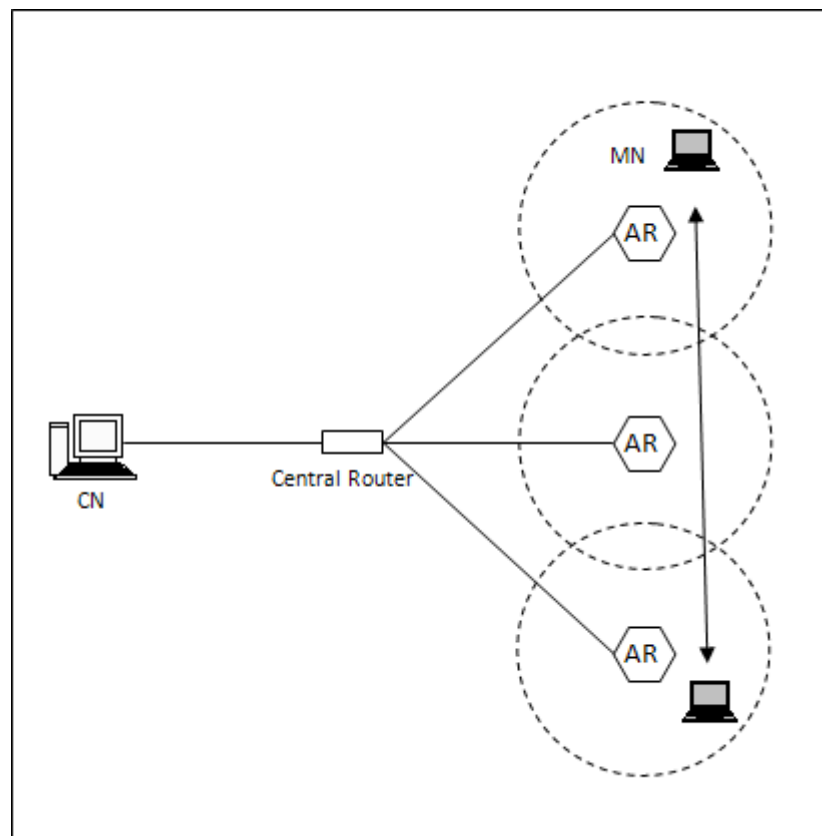
This phase will conclude the study and the results achieved will be stated together to show the final findings. This phase will also give some ideas to areas of improvement to reduce the latencies experienced during handoff in MIPv6 when using Route Optimization scheme.



**Figure 3.1:** Operational Framework.

### 3.3 The Network Model

The simulation network uses different wired and wireless equipment. The wired back bone of the network is a 100Mbps Ethernet LAN. The wired LAN is connected through routers and divided to three networks each one represents a different IP subnet. Each network is connected to an AP that works as an Access Router (AR). The distance between the ARs is 450m and the transmission range is set by default to 250m. Thus, the coverage areas of the AR overlap. Figure 3.2 shows the proposed simulation network model.



**Figure 3.2:** Simulation Network Model.

The Network Model represents a simple and popular network environment. This Model can be used within university campuses or companies where security will be set to the same level, allowing the MNs to roam without the hassle of overcoming security issues.

### **3.4 The Simulation Model**

For simplicity and to get the exact results from each simulation, there will be one CN that communicate with one MN. The MN will change its network attachment as it moves from one AR to another. The MN will experience times where it will be within two ARs range at the same time. Later other MNs and CNs will be added to experiment more realistic scenarios. The proposed approach will be tested using the same scenarios and the results will be compared with existing Route Optimization handoff schemes. The Link Delay (LD) between the Central Router (CR) and the ARs will be set to (1.8ms). The LD between the CR and CN will be set to (10.8ms) in order to simulate a crowded network connection (Perez-Costa et al., 2003a).

TCP and UDP packets will be transferred between the CN and the MN. The VoIP streaming is considered a very good comparison metric to compare the different handoff schemes; Therefore, UDP packets with specified size and interval time will be used to resemble VoIP streaming (Jung et al., 2003)

### 3.5 The Simulation Setup

The scenario of this simulation resembles a simple wireless LAN hotspot deployment that can be found within university campus, for example. It comprises of three Access Routers (ARs), with distances of 450m between each AR. The transmission range is set by default to 250m in ns2, which will create overlapping coverage areas. The ARs work according to the IEEE 802.11b wireless LAN standard. In order to provide realistic results out of the simulations, two scenarios will be simulated using the same Network Model and Simulation Model. To study the efficiency of each scheme, the first scenario will simulate one MN communicating with one CN. This will provide information how the scheme works with minimum collisions and minimum shared AR queues. See Appendix (B).

The second scenario will represents a more realistic situation where more mobile and corresponding nodes will share the network medium and MNs share the AR queue. The MNs will receive packets from the shared AR queues and compete to get access to the AR channel. This scenario also illustrate how well the scheme perform when more than one MN perform a handoff at the same time and with more collisions. See Appendix (C).

During both scenarios different input files will be injected using a packet generator embedded within ns2 as simulation media. Firstly, files will be transferred between the MN and CN over the transport protocol (TCP). Simulating an endless FTP source will help understand the impact of changing IP attachments on the congestion control mechanism of TCP. This can be seen in Appendix (B) in the code:

```
#####
# Create FTP over TCP traffic
#####
proc set-ftp { } {
    global ns cn_ mobile_ sink
    set tcp [new Agent/TCP]
    $ns attach-agent $cn_ $tcp
    set sink [new Agent/TCPSink]
    $ns attach-agent $mobile_ $sink
    $ns connect $tcp $sink

    set ftp [new Application/FTP]
    $ftp attach-agent $tcp

    $ns at 1.0 "$ftp start"
}

```

**Figure 3.3:** TCL Code for Generating FTP over TCP Traffic in NS-2.

Secondly, real-time media will be simulated to represent VoIP. The User Datagram Protocol (UDP) provides constant traffic where no acknowledgements required. This kind of traffic is usually generated by real-time applications (Perez-Costa et al., 2003a). The VoIP packets will be resembled by a (CBR) source, producing fixed length packets of 200Bytes. This will represent a payload of 160Byte and a 40Byte header (RTP+UDP+IP). The packets will be sent at a rate of 64Kbps (Jung, et al., 2003).

```
#####
# Create CBR over UDP traffic
#####
proc set-cbr { } {
    global ns cn_ mobile_ sink
    set udp [new Agent/UDP]
    $ns attach-agent $cn_ $udp
    set sink [new Agent/LossMonitor]
    $ns attach-agent $mobile_ $sink
    $ns connect $udp $sink

    set cbr [new Application/Traffic/CBR]
    $cbr attach-agent $udp
    $cbr set packetSize_ 200
    $cbr set rate_ 64Kb

    $ns at 1.0 "$cbr start"
}

```

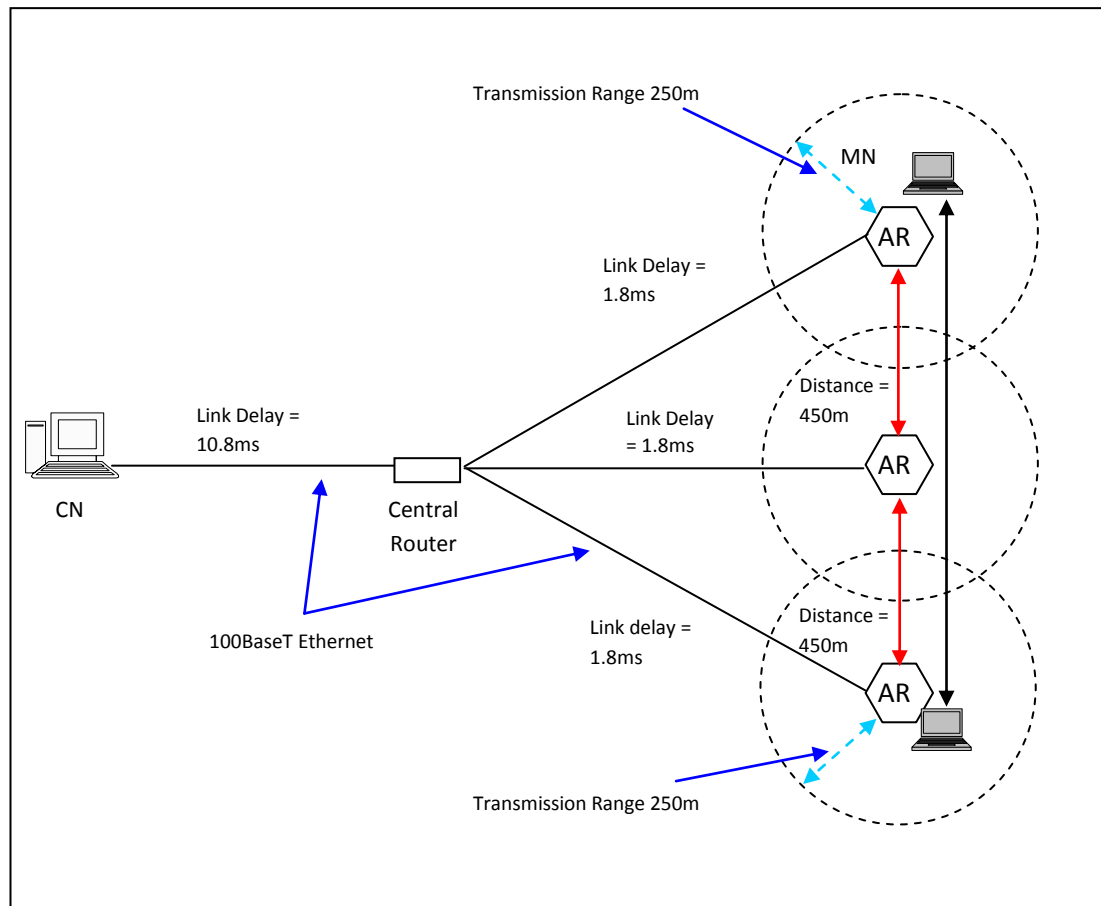
**Figure 3.4:** TCL Code for Generating CBR over UDP Traffic in NS-2.

Two main simulation scenarios will be simulated. Table 3.1 shows the two different scenarios. Scenario 2 will be divided to 3 scenarios where each scenario increases the number of simulated MNs and CNs by 10.

**Table 3.1:** The Simulation Scenarios

	Scenario 1	Scenario 2
No of MN	1	10 / 20 / 30
No of CN	1	10 / 20 / 30
MN Moving Speed	10m/s	Random
MN Moving Pattern	Deterministic Path	Random Waypoint Algorithm
Link Delay (CN to CR)	10.8ms	10.8ms
Link Delay (CR to AR)	1.8ms	1.8ms
Simulation Time	600s	600s
No of Handoffs per MN	6	3-7
Simulated Protocols	TCP / UDP	TCP / UDP

As a wireless medium the 2Mbps Wireless LAN 802.11 DCF provided by ns2 will be used. The access routers use the same frequency band since no roaming process is standardized for 802.11 and thus, roaming protocols are proprietary. The wired backbone is a 100Mbps Ethernet LAN. The Link Delay (LD) between CR and the ARs is set to 1.8ms and between CR and CN is set to 10.8ms in order to further resemble a crowded network environment (Perez-Costa et al., 2003a). Figure 3.5 shows the Simulation Model.



**Figure 3.5:** Simulation Model (Perez-Costa et al., 2003a)

### 3.5.1 Simulation Metrics

To analyze handoff schemes performances these aspects will be studied for each MN; handoff latency, packet losses during handoffs, signaling load and obtained bandwidth per station. These parameters are explained with more detail in Table 3.2.

**Table 3.2:** Simulation Measurement Metrics.

<b>Metric</b>	<b>Description</b>
Handoff Latency	The time that elapses between the last packet received via the old route and the arrival of the first packet along the new route after a handoff.
Packet Loss	The number of packets lost during handoffs.
Signaling Load	The total number bytes of Binding Updates and Binding Acknowledgements messages sent during handoffs.
Bandwidth per Station	The number of Megabytes or Kilobytes per second received by the receiving node.

### **3.6 Simulation Test-Bed**

The simulation will be run using a PC equipped with a Pentium 3 processor running at speed of 1.0 GHz. The PC also equipped with SiS 630 graphic card and SiS 900 series Ethernet card. The PC is running a Redhat Linux 7.3 and the scenarios will be simulated using ns2.1b6 MobiWAN extension.

### **3.7 Summary**

This chapter describes the method in which the whole project will be carried out. The Operational Framework is where the different phases and steps which will be used to achieve the objectives of this project are explained. The ns2 simulator is considered one of the most favorable simulators among network technologies

researchers. The Simulation model with its different scenarios will help to illustrate the differences between the different schemes when simulated under the same environment.

The Network Model shows the design of the simulation network where all the different Route Optimization handoff schemes will be tested. The network model will not change throughout the study to maintain the creditability of testing different schemes on the same network.

The Simulation Model briefly describes the parameters of the simulation. The parameters will change according to different scenarios. The number of MN and CN will change to simulate real situations. The simulation Model will focus on low security scenarios since security issues are out of the scope of this project.

The metrics of the scenarios will help studying the actual performance of TCP and UDP over MIPv6 during handoffs. The simulation model, setup and test-bed have been described in this chapter to give a better picture of how the simulations were performed.

## **CHAPTER 4**

### **EARLY HANDOFF (EH)**

#### **4.1 Introduction**

The Mobile Internet Protocol version 6 (MIPv6) has been proposed since the year 1996 when Perkins, C. and Johnson, D. B. first proposed the protocol. There are several proposals that try to improve the efficiency, reduce latencies and fix other issues related to mobility support. When the main protocols in TCP/IP suite, the Transmission Control Protocol (TCP) and Internet Protocol (IP) were first implemented, the computer networks were using wires to connect large non-movable nodes with each other. The IP which is used to route packets according to logical addresses does not support nodes mobility. Whenever a computer changes its point of attachment, its logical address is changed too and the connection itself is considered lost. MIPv6 allows connected sessions to remain active even if the logical address of a computer has changed. However, that improvement in mobility support was achieved at the cost of the time and routes packets use to reach their destinations.

Long handoff latencies actually cause routers to assume that the physical connection to a certain node has been lost. In that case the routers discard all packets meant for that node. Handoff latencies also have a bad impact on transmission protocols' performance. TCP is engineered to serve wired connections and avoid transmission degradation caused by a limited number of collisions or by the limited size of routers' buffers. Using the air as a transmission medium has introduced new causes of TCP and UDP transmission degradation. High Bit Error Rate and higher possibility of collisions meant more problems facing packets delivery. That can have a bad impact on the active sessions and lousy users' experiences.

## **4.2 Early Handoff**

One way to improve the performance of MIPv6 is to reduce the time for a mobile node to change its logical address (IP). Every Mobile Node (MN) has to go through several steps to perform a handover. One of the first steps requires a MN to wait for the loss of three consecutive Router Advertisement (RA) messages from its current Access Router (AR). That means the MN has to wait for a period of time, which can go up to around 1 second before it assumes that it has lost its connection with its current AR. That is the case even though a MN receives RAs from a new AR in coverage overlapping areas. The current MIPv6 specification document (Johnson, Perkins, and Arkko, 2004), requires the MN to wait for the loss of connection with its current AR before it performs a handover procedure. The Early Handoff (EH) uses the overlapped coverage areas to perform a handover without waiting for the loss of communication with the current AR. In most cases when a MN enters an overlapping coverage area that means it is actually traveling from the coverage of one AR to another AR. EH technique assumes that this is always the case and starts performing the handover procedure whenever a RA from a new router has been received. Therefore, The MN does not have to wait until it loses its link with the

current AR to perform a handoff thus saving some valuable milliseconds of disconnection that may cause an active session to degrade or end.

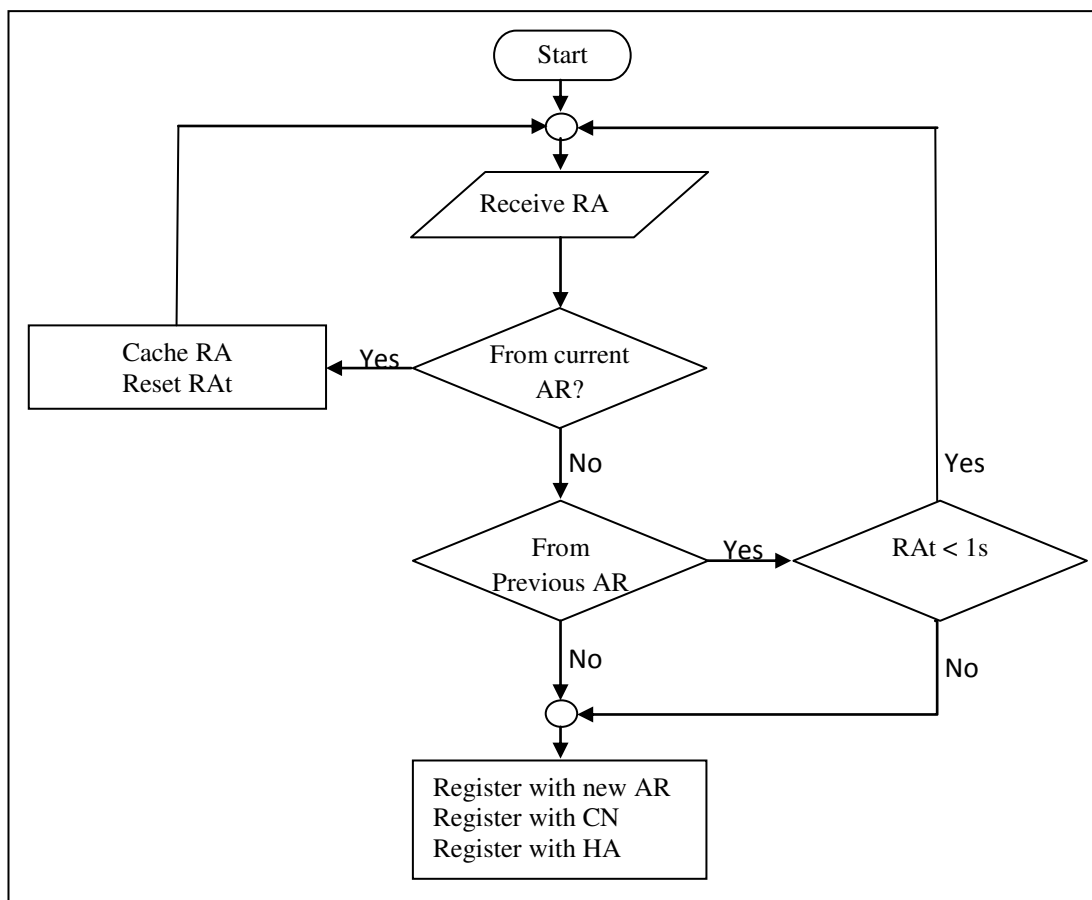
The Early Handoff scheme will use the current specification of MIPv6 stated in (Johnson, Perkins, and Arkko, 2004). Whenever a MN receives a RA it checks if this RA is from a new AR. If that is the case the MN assumes that it is moving and entered an overlapped area or it has already gone out of its previous AR's coverage area. The MN checks if the RA is from the previous AR to help eliminate the unnecessary re-registration with its previous AR while still traveling through the overlapping area. In the case where the MN enters to an overlapping area and then goes back to its original AR, the MN will use the timer specified to tell if it had lost three consecutive RA from its current AR. If that happened the MN will perform a normal handover procedure.

The algorithm bellow describes the steps a mobile node has to follow when using the proposed technique. When a MN receives a RA, it performs the following steps:

1. The MN checks if the RA belongs to the current AR.
  - a. Yes: the MN caches the message, reset RA timeout counter (here called RA<sub>t</sub>) and goes back to the beginning.
  - b. No: goes to step 2.
  
2. The MN checks if the RA belongs to the previous AR.
  - a. Yes: the MN checks if there are RAs from the current AR less than 1s old (RA<sub>t</sub> < 1s)
    - (i) Yes: the MN is still in overlapping area and had just performed a handover. Backs to the beginning.
    - (ii) No: the MN entered an overlapping area and then went back to its previous AR. Goes to step 3.
  - b. No: the MN is going to enter a new AR area. Goes to step 3.

3. Perform a MN handover as following:
  - a. Registers with the new AR.
  - b. Sends Binding Update to the Corresponding Node.
  - c. Sends Binding Update to the Home Agent.

Figure 4.1, illustrates the flowchart of the (EH) algorithm.



**Figure 4.1:** Early Handoff Scheme Flowchart.

### 4.3 EH Implementation

EH can be implemented with any existing handoff scheme to increase the performance during handoffs. Therefore, existing features and codes of other schemes can be used. The programming is done by adding new commands and codes to the existing (mipv6.cc) file, see Appendix (A). The new code is added within existing methods and reused some existing variables.

#### 4.3.1 Router Advisements Beacon

Firs thing is to set the Base Station's beacon time to 500ms. BS beacon is actually a random time used by each BS to broadcast RAs. That is shown in Figure 4.2, where the changes in the original code had been underlined.

```
MIPv6Agent::MIPv6Agent() : Agent(PT_UDP), bcast_target_(0),
timer_(this), beacon_(0.500), print_info_(0)
```

**Figure 4.2:** Set BS Beacon to 500ms.

#### 4.3.1 Adding the EH Code

Figure 4.3 shows where the first part of the code should be added. The method `recv_ads(Packet *p)` is triggered whenever a MN receives a new RA.

```
void MNAgent::recv_ads(Packet *p)
```

**Figure 4.3:** The Method Where the Code Will Be Added

When a MN receives a RA it checks whether this RA is from a known BS or not. This is done by checking the two top entries in the BS list; the BS on the top is the current BS which the MN is connected through at the moment. The second in the list is the previous BS which the MN has just left. The code added is as illustrated in Figure 4.4. The first line points to the head of BS list. This should be the one which MN is currently attached to. The second line reset the pointer to point to the second entry in the list.

```
Entry *node1 = head(bslist_head_);
node1 = lookup_entry(iph->saddr(), node1->next_entry());
```

**Figure 4.4:** Set a Pointer to the Second BS in the BS List.

As Figure 4.5 shows, if the RA is from the current BS, reset its expiration time and set the RA<sub>t</sub> to NOW. Where NOW is the current time.

```
RAt = NOW;
```

**Figure 4.5:** Router Advertisement Timer.

If the RA is not from the current BS then check if it is from the previous BS. To eliminate the possibility of bouncing between BSs within overlapped area, the MN checks if it lost three consecutive RAs.

```
} else if (node1 && ((NOW-RAt) < 1.0)) {
}
```

**Figure 4.6:** Checking If the Timer Is More than 1 Second.

If the RA is from the previous BS and the RA had been reset within the last one second then do nothing.

In other cases two possibilities may occur. The first possibility is when a MN had traveled within an overlapped area and went back to the previous BS without leaving the overlapped area. The second possibility is when the RA is from a new BS. In such cases the MN should register with the BS which sent the received the RA. The BS which sent the last RA will be saved at the top of the BS list as the most current one. The MN has lost its care-of-address and a registration method is called. Notice that the code illustrated in Figure 4.7 has not been tempered with and is the original code from the (mipv6.cc) file (Ernest, 2001).

```

else {

    // New ads. This BS is not yet recorded in our BS list
    // Record it in front of list. This BS will become current BS
    // We need a new COA on the BS's subnet and we need
    // to register BS in list and new binding with HA / CN.

    if (print_info_) cout << " *** new BS ***";
    Entry *pbs = add_bs(iph->saddr());

    // XXX: I should send a request to obtain a COA
    pbs->update_entry(NONE, NONE, get_coa(iph->saddr()), NOW, rh-
>lifetime());
    coalost_ = TRUE;
    reg();
}

```

**Figure 4.7:** A New Router Advertisement Has Been Received.

The second part of coding EH is to change the way registration requests are sent. The original MIPv6 documentation states that the MN should first register with HA, waits for acknowledgement and then registers with CN. This will be changed so that the MN sends the registration request at the same time. The MN would not wait to receive registration acknowledgement from HA, before proceeding with CN registration, because that would increase the handoffs delay.

Figure 4.8; shows the method reg() where the registration procedures and priorities will be changed.

```
void MNAgent::reg()
```

**Figure 4.8:** Method for Changing the Registration Priorities.

The registration sequence will be changed to become as shown in Figure 4.9. The Mobile Node shall send Binding Updates to the Corresponding Node and the Home Agent without waiting for any acknowledgements. However, this could increase the number of lost packets in the case the registration failure with HA.

```
if ( rt_opti_ ) send_standard_bu(BU_CN);  
send_standard_bu(BU_HA);
```

**Figure 4.9:** Registering a MN with CN then HA.

#### 4.4 Summary

Since its proposal, MIPv6 is facing different implementation issues and problems. The Route Optimization reduces the time needed for packets to travel from CN to MN. On the other hand, long handoff latencies are degrading the throughput of TCP and UDP over MIPv6 while a MN is experiencing a handoff. Many extensions and schemes are proposed to solve the problems of long handoff latencies.

The Early Handoff scheme discussed here is a new MIPv6 handoff scheme proposed in order to take advantage of the new RAs that a MN receives every time it enters an overlapped area. These messages can indicate a MN movement and the possibility of handoff in near future. In EH a MN does not wait until it loses its connection with its current BS in order to perform a handoff. This is thought to give a better performance to both TCP and UDP connections.

The idea of EH and the algorithm were described and the programming needed in order to achieve these modification were also listed in this chapter.

## **CHAPTER 5**

### **SIMULATION RESULTS**

#### **5.1 Introduction**

To achieve the aim of this project, which is to propose an alternative fast handoff scheme for MIPv6, previous works by other researchers are studied. Each approach proposed before has its advantages and limitations. Some schemes work fine within certain environments or scenarios, while others fail to improve performance in such scenarios. Different schemes have been tested in simulation environment using the same network environment to illustrate how well each scheme performs during Route Optimization handoffs.

The simulation is divided into different scenarios representing different number of communicating nodes. Each scenario simulates Transmission Control Protocol (TCP) and User Datagram Protocol (UDP) packet fragments. Although TCP provides reliable transferring protocol, it also suffers from congestions that can force slow starting and window sizing strategies, thus, reduce the throughput of the protocol. The UDP on the other hand, does not require acknowledgements. While

that help making it better used for real-time applications, UDP is actually considered an unreliable transmission protocol.

This chapter discusses the simulation environment which was used in order to achieve the goals of this project. The results of the compared schemes explained and discussed here as well.

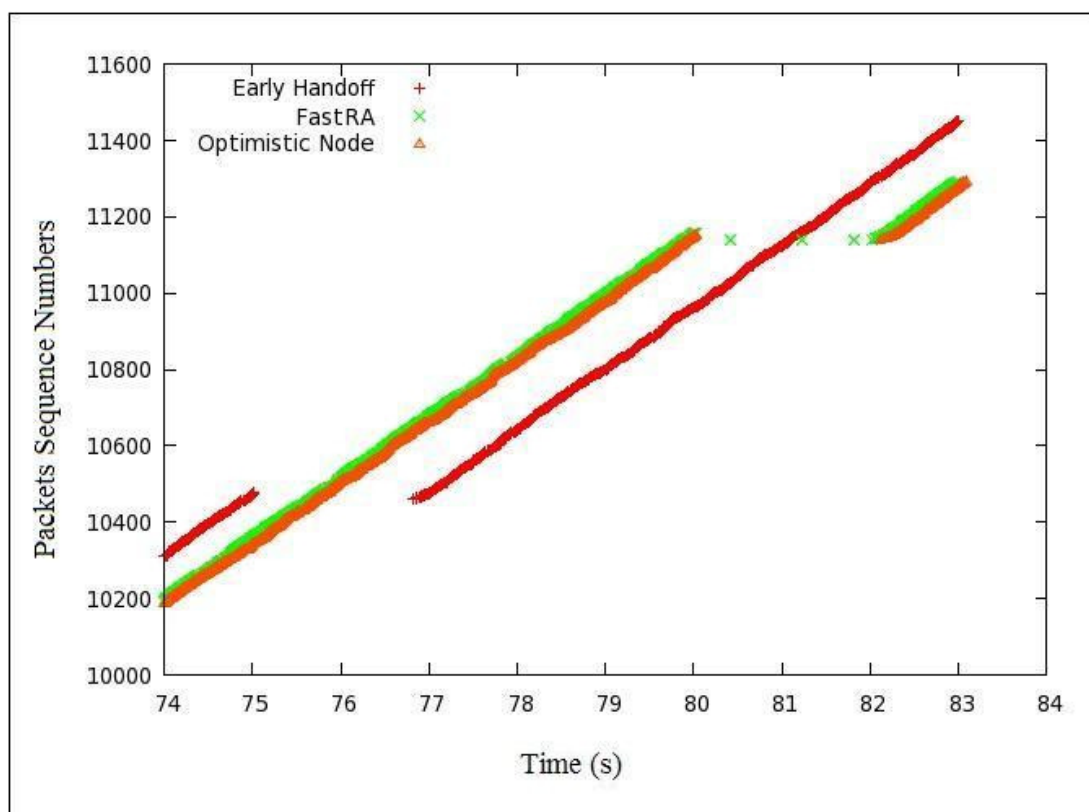
## **5.2 Simulation Results**

The Early Handoff scheme was simulated and compared with FastRA and Optimistic Node handoff schemes. Since the ns-2 does not have a real implementation of IP addressing and for the lack of support to DHCP and Neighbor Discovery in MobiWAN, the simulation of A-DAD and O-DAD were not simulated. In addition the RA-Caching scheme was also not simulated because it required changing the simulation model.

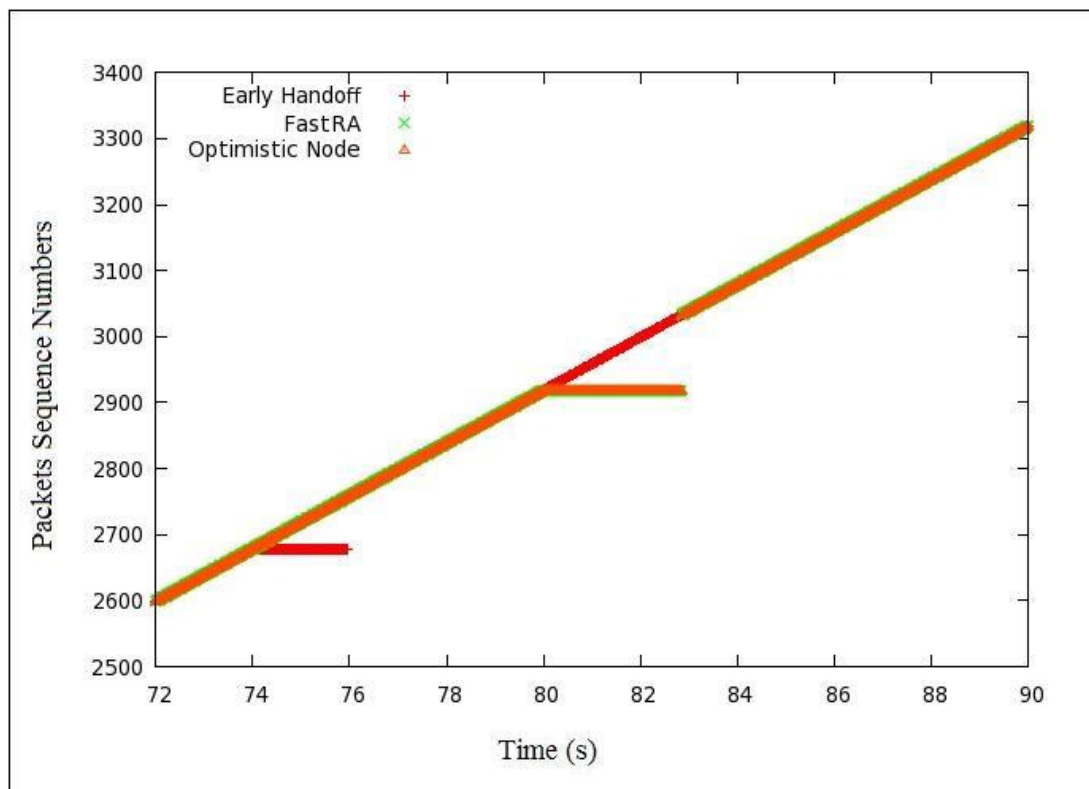
### **5.2.1 Handoff Latency**

The time that elapses when a MN perform a handoff is very crucial. Many delay sensitive applications and protocols suffer greatly if this time passes a certain limit. TCP is an interruption sensitive protocol since it requires the receiving node to acknowledge received packets. Failing to receive an acknowledgment for a packet means that the packet will be resent after a certain time. Other protocols such as UDP do not provide such reliability, which makes UDP unaware of MN handoff. Any packets sent during the handoff will be dropped by the router without alerting the

sending node. However, some delay sensitive applications such as VoIP or video conferencing can suffer from long MN handoffs. Figure 5.2 illustrates that UDP protocol was not aware that sent packets were not actually delivered and it did not resend them. On the other hand TCP will try to resend all the un-acknowledged packets. The handoff latency was plotted using GNUPlot. In the TCP case, the bytes which were received by the Base Stations were plotted since the trace file contained data of the wired portion only. The CN stop sending packets when the timeout for ACKs of sent packets are expired. The ns agent (TCP) uses cumulative acknowledgement policy; therefore the CN resent some of the previously sent and unacknowledged TCP packets. In UDP case the agent (LossMonitor) is used to check the number of the last packet received by the MN and first received after the handoff had finished.



**Figure 5.1:** TCP Packets Interruption during Handoff.

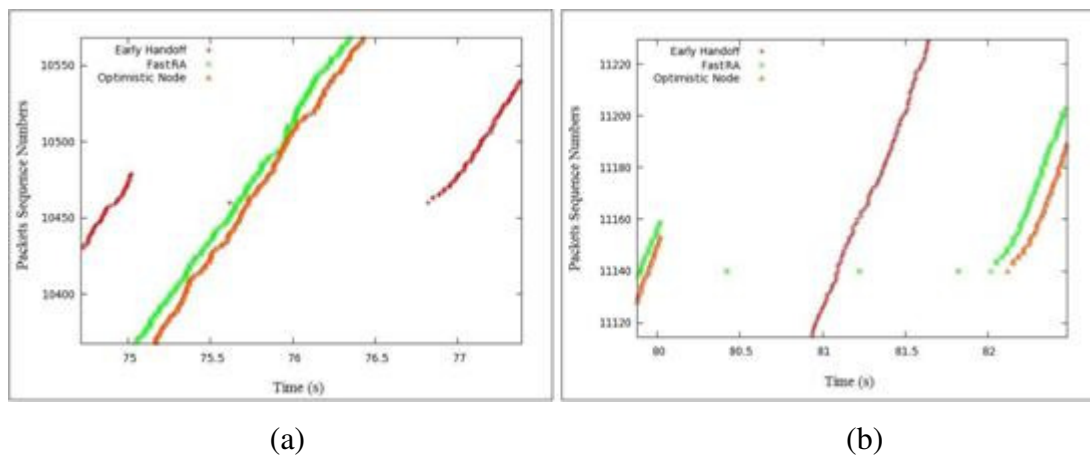


**Figure 5.2:** UDP Packets Interruption during Handoff.

Figures 5.1 and 5.2 clearly show that compared to FastRA and Optimistic Node, the Early Handoff scheme performs a much earlier handoff than the other two. That can give it a little advantage when transferring TCP packets.

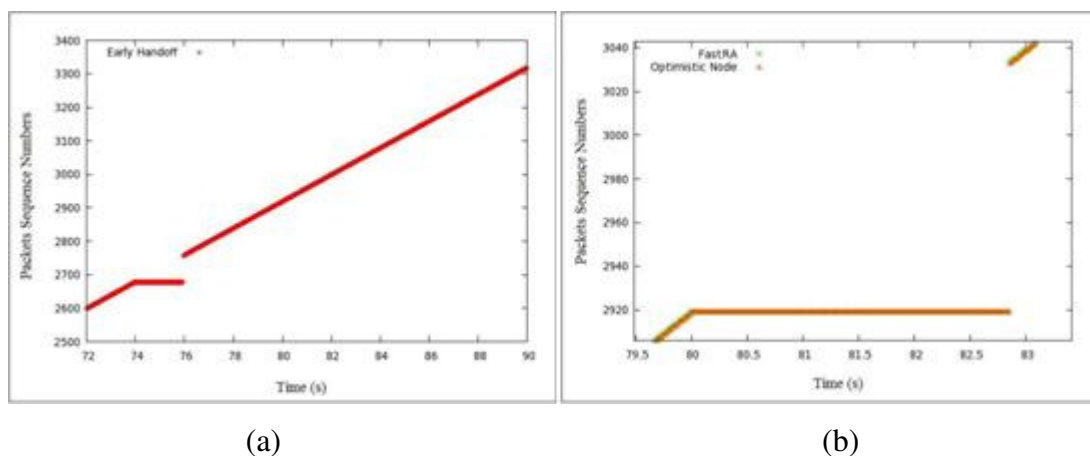
From the Figure 5.3(a) it is clear that TCP performs better using EH during the single MN simulation. The handoff happens about 5 seconds before FastRA and Optimistic Node at exactly at the simulation time of 75.01 seconds. Also, since the MN does not wait for connection interruption before performing a handoff, the duration the handoff takes about 1.7 seconds. On the other hand, Figure 5.3(b) shows that FastRA and Optimistic Node handoffs take about 2 seconds each. It is also clear that FastRA performs a little better than Optimistic Node probably because the latter keeps performing unnecessary Home Registration procedures periodically. The shorter handoff time and early handoff help improve the performance of the TCP protocol during the handoff. There are some limitations in the ns-2.1b6, such as while using the TCP protocol the trace file did not include any information regarding

the MN. For that reason the data acquired using UDP simulation would be more accurate.



**Figure 5.3:** TCP Interruption in EH (a), FastRA and Optimistic Node (b) (1 MN).

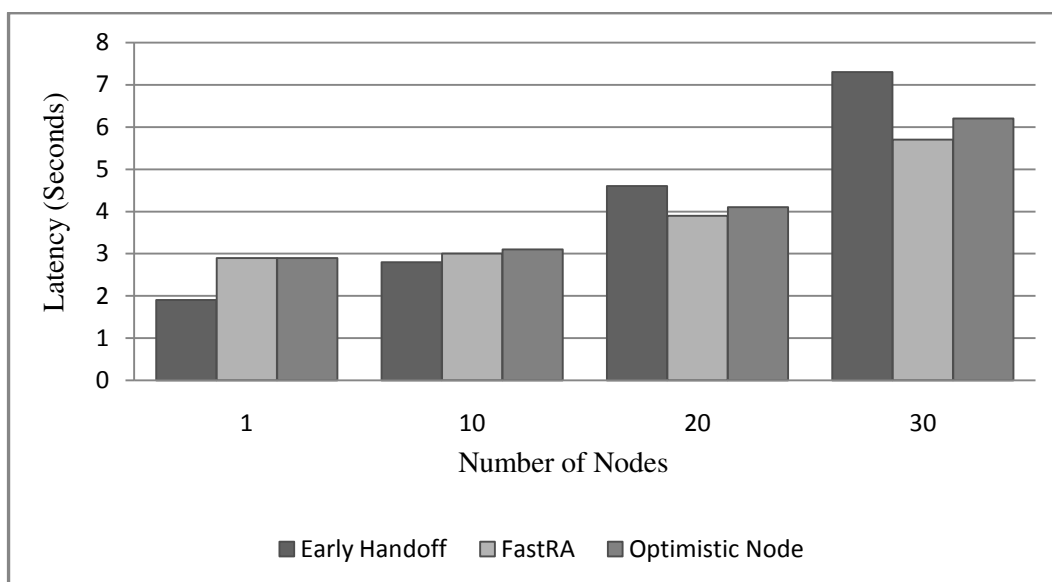
Figure 5.4(a) shows that when simulating UDP packets, the EH performs much better than FastRA and Optimistic Node. The handoff occurs about 6 seconds earlier than FastRA and Optimistic Node at the simulation time of 74 second, and takes just 2 seconds. FastRA and Optimistic Node perform handoffs around the time 80 seconds, as illustrated in Figure 5.4(b), and take about 2.9 seconds to perform a complete handoff. Since in UDP un-delivered packets are not resent, the CN keeps sending packets periodically without concerning about the handoff.



**Figure 5.4:** UDP Interruption in EH (a), FastRA and Optimistic Node (b) (1 MN).

From Figures 5.1, 5.2, 5.3 and 5.4 it is clear that when simulating a single MN, EH performs better than FastRA and Optimistic Node in both situations. However these simulations still miss the real time that would be contributed if addressing schemes were used.

Figure 5.5 illustrates the impact of number of communicating nodes on the handoff latencies. 10, 20 and 30 nodes were simulated using the same network model as before. The performance of the three handoff schemes displays how more communicating nodes negatively affect the handoff latencies.



**Figure 5.5:** Impact of Number of MN on the Handoff Latency.

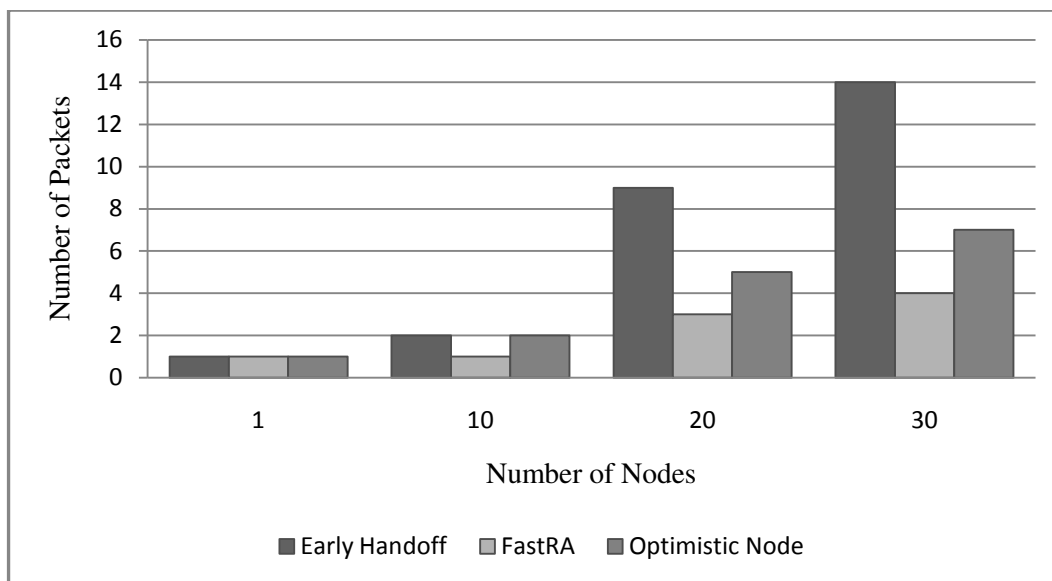
While EH shows promising performance in the single MN scenario, all that change when the number of communicating nodes increased and a random movement pattern was applied. Sometimes unnecessary handoffs happen whenever a MN receives a new RA. That adds a heavy load on the channel used to send the data. As seen in Signaling Load, EH and Optimistic Node both send and receive unnecessary BU messages. Such messages with the increased number of handoffs in EH increase the handoff latency dramatically. Figure 5.5 shows that in EH the single handoff latency increases from just less than 2 seconds to reach around 7.3 seconds

for a single handoff in 30 nodes scenario. The Optimistic Node latency also increases from 2.9 to 6.2 seconds, probably because of performing the Home Registration periodically. The FastRA latency increases from 2.9 to 5.7 seconds making it the best handoff scheme in this situation.

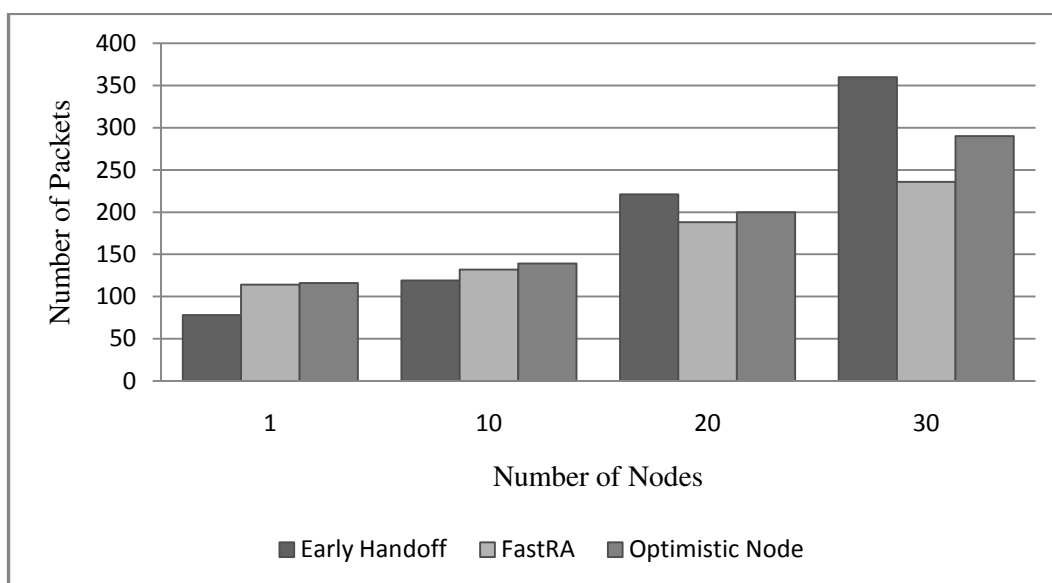
### **5.2.2 Packet Loss**

During a handoff the MN loses all connections with any CN for a period of time. Using TCP the CN sends packets with sequence numbers to the Central Router which in turn forwards them to the Base Station. The Base Station broadcasts the packets and waits for acknowledgments from the receiving MN to send them back to the CN. UDP, on the other hand, does not share that reliability feature with TCP. Therefore it has a higher packet loss rate.

Figures 5.6 (a and b) illustrate the differences between TCP and UDP as transport protocols and the impact of the number on the packet loss rate.



(a)



(b)

**Figure 5.6:** Packet Loss Rates in TCP and UDP Related to Number of Nodes.

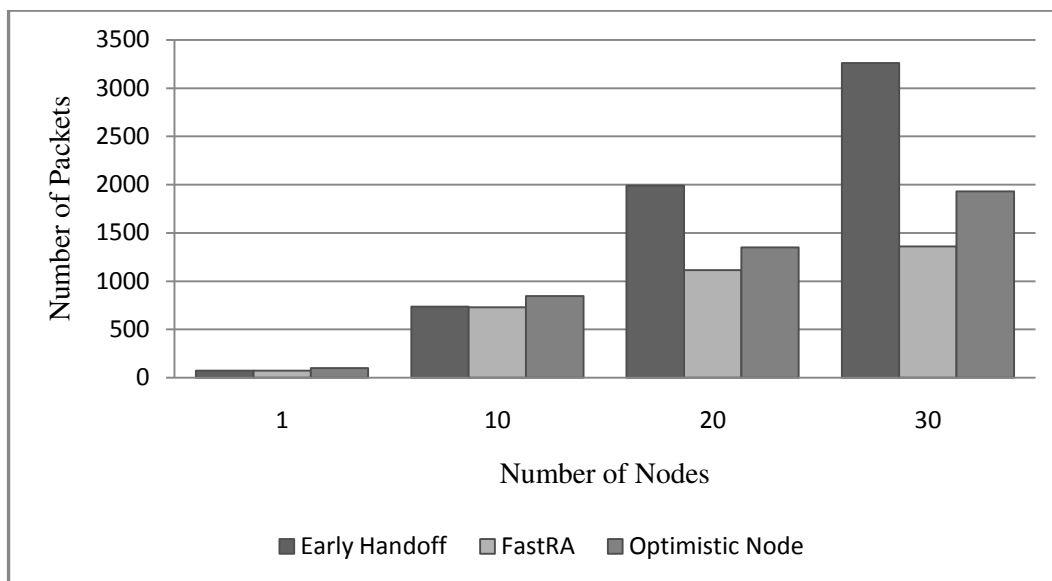
When using TCP if a CN did not receive ACKs for sent packets, it resends the packets after a timeout. That helps reduce packet loss rate in TCP when just two nodes are communicating. In single CN and MN scenario the three schemes have given the same result of 1 lost packet. However, increasing the number of the nodes resulted in higher rates of packets loss. In EH the number of packets lost increased to

reach 2, 9 and 14 packets in the 10, 20 and 30 nodes scenarios respectively. The Optimistic Node had an average rate of 7 packets lost and the FastRA just 4 packets in the 30 nodes scenarios. The degradation in performance also continued in UDP scenarios where EH had an average of 78 packets lost during a single handoff in single MN scenario. That number increased to reach about 119, 221 and 360 packets lost during a single handoff in the 10, 20 and 30 nodes scenarios. Optimistic Node increased from 116 packets lost to 290 packets. On the other hand, FastRA scored from 114 to 236 packets lost. The agent (LossMinitor) was used to trace the number of packets received by the MN. Later the number of lost packets was calculated using an AWK script.

### **5.2.3 Signaling Load**

Signaling Load measures the total number of Binding Updates and Binding Acknowledgments that were sent through the wireless channel. These packets are consisted of just 40Bytes. Figure 5.7 shows the number of BUs and BAs that had been sent throughout the simulation time of 600 seconds. The number of BUs and BAs received by the BSs is extracted from the trace file using the command (grep) that included with Linux operating system.

EH and FastRA shared the same signaling load when simulating a single node both sending 72 packets of BUs and BAs. Optimistic Node sent 98 packets since it performs additional home registration tests.



**Figure 5.7:** Signaling Load Vs Number of Nodes.

EH and FastRA send BUs and BAs during handoffs right after acquiring a CoA. Optimistic Node, on the other hand, sends these packets periodically even if a MN never goes through a handoff.

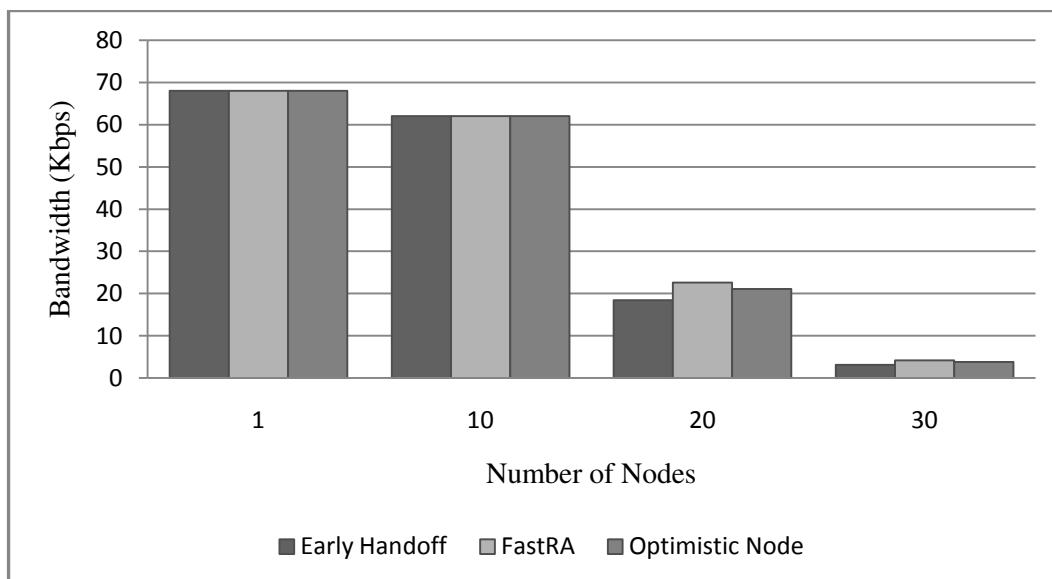
When multiple nodes were communicating at the same time the EH puts more load on the channel than the other two schemes. That is clear in the 20 and 30 nodes scenarios where the signaling load reached 1990 and 3260 respectively. Such loads contribute to the degradation of bandwidth and packet loss rates.

#### 5.2.4 Bandwidth per Station

The Bandwidth per Station is computed by measuring how many Kilo Bytes per second did the MN receive. In this project the bandwidth was measured in UDP scenarios only mainly because it is easier to isolate the receiving node from the

others. This was done using the (LossMonitor) agent in ns-2. Each second the number of bytes received by the MN were calculated and multiplied by (8) then divided by (1024).

Figure 5.8 illustrates the average bandwidth received by a MN node in 4 different scenarios. The UDP packets sized only 200 Bytes and were sent at a rate of 64Kbps. Figure 5.8 shows that the three schemes delivered the same performance of about 68Kbps in a single node scenario.



**Figure 5.8:** Bandwidth per Station.

In the first two scenarios it is clear that the signaling load did not have a severe effect on the bandwidth per station since the transmission medium bandwidth was much higher. When the number of nodes increased to 30, the bandwidth degrades drastically to reach 3.1Kbps, 4.2Kbps and 3.8Kbps for EH, FastRA and Optimistic Node respectively.

### 5.3 Discussion

Tables 5.2, 5.3, 5.4 and 5.5 summarize the results of the simulations and give a better idea about the performance of each scheme in each scenario. It is clear that the proposed handoff scheme (Early Handoff) performs better in terms of throughput, packets loss rate and handoff latency in scenarios with little number of nodes. The EH assumes that any MN which enters an overlapping area will perform a handoff. However in some cases a MN may enters an overlapping area and stop there or it may enters an overlapping area and then returns back to the same BS. In such cases if a MN is using the EH it will perform unnecessary handoffs. Since the MN does not actually lose its previous point of attachment, going through a handoff means more packet loss and degradation in throughput. Such negative effects are not necessary in these cases. Table 5.1 is the proposed criteria on which the handoff schemes were compared upon.

**Table 5.1:** Criteria of Measurement.

	Performance				
	Excellent	Good	Bad	Worse	Worst
Handoff Latency	0.0 – 2.0	2.1 – 3.5	3.6 – 5.0	5.1 – 6.5	6.6 – 8.0
Packet Loss	0 – 1	2 – 3	4 – 6	7 – 8	9 – 15
Signaling Load	0 – 90	91 – 1000	1001- 1500	1501 – 2000	2001 - 4000
Bandwidth per Station	55.0 – 100	40.0 – 54.9	25.0 – 39.9	10.0 – 24.9	0.0 – 9.9

**Table 5.2:** Single Node Scenario Performance Comparison Results.

	Early Handoff	FastRA	Optimistic Node
Handoff Latency	Excellent	Good	Good
Packet Loss	Excellent	Excellent	Excellent
Signaling Load	Excellent	Excellent	Good
Bandwidth per Station	Excellent	Excellent	Excellent

**Table 5.3:** Ten Nodes Scenario Performance Comparison Results.

	Early Handoff	FastRA	Optimistic Node
Handoff Latency	Good	Good	Good
Packet Loss	Good	Excellent	Good
Signaling Load	Good	Good	Good
Bandwidth per Station	Excellent	Excellent	Excellent

**Table 5.4:** Twenty Nodes Scenario Performance Comparison Results.

	Early Handoff	FastRA	Optimistic Node
Handoff Latency	Worse	Bad	Bad
Packet Loss	Worst	Good	Bad
Signaling Load	Worse	Bad	Bad
Bandwidth per Station	Worse	Worse	Worse

**Table 5.5:** Thirty Nodes Scenario Performance Comparison Results.

	Early Handoff	FastRA	Optimistic Node
Handoff Latency	Worst	Bad	Worse
Packet Loss	Worst	Good	Worse
Signaling Load	Worst	Bad	Worse
Bandwidth per Station	Worst	Worst	Worst

The amount of bytes used by the BAs and BUs can also negatively affect the throughput and bandwidth. Since sending and receiving unnecessary BUs and BAs means larger signaling load and a higher risk of collisions. Using EH in a crowded environment with random movement patterns increases the rate of handoffs. That would increase the signaling load and packet loss rates and would cause the performance of both TCP and UDP connection to degrade severely as can be seen in Tables 5.4 and 5.5. Optimistic Node should have had smaller handoff latency than

FastRA, nevertheless, the simulations proved that FastRA had performed better. Actually the performance of FastRA depends on which addressing scheme it uses. FastRA does not implement a specific addressing scheme to go with it, on the other hand Optimistic Node states that O-DAD addressing scheme should be used during handoffs. That may help Optimistic Node to perform better than FastRA in real world.

Bear in mind that the simulations environments were duplicated address free. Also during the simulations security and authentication problems were not addressed although they actually contribute highly in the performance of handoff schemes. At such environments Optimistic Node should perform better since it registers with the home agent periodically. The periodic home registration help eliminate the possibility of home registration failure which is possible if authentication is applied.

In conclusion the study shows that different wireless scenarios require different handoff schemes to help reduce the negative effects of MIPv6 handoffs.

#### **5.4 Summary**

The EH, FastRA and Optimistic Node were simulated using the simulation model described in Chapter 3 with the scenarios described in this chapter. From the simulations it was clear that compared with the other handoff schemes, the EH was able to achieve better performances under certain conditions and scenarios. The simulation metrics showed clearly what are the advantages and limitations of the three simulated handoff schemes, EH, FastRA and Optimistic Node. However, the simulation results were achieved in an environment free of addressing and security problems which can affect the performances of each scheme in real implementation.

## **CHAPTER 6**

### **ACHIEVEMENTS, LIMITATIONS AND FUTURE RECOMMENDATIONS**

#### **6.1 Introduction**

Computer networks are increasing in number every day. With technology advances, computers have become smaller and much faster. An average person can now own a small PC and a handphone in most of the developed countries. More users are demanding to have the ability to connect to the existing computer networks either at work, home or pleasure places. Mobile IP (MIP) was the solution for users demanding for Internet connection while moving. It allows small portable PCs including handphones to roam between hotspots. However, due to the mobility of the hosts, the transmission sessions between nodes are subject to interruptions. A handoff occurs when a mobile node or host changes its point of attachment or Base Station. Base Stations (BSs) or Access Routers (ARs) are wired routers with built-in Access Points (AP). Changing point of attachment means changing the IP address, which is the back-bone of transmission protocols. All packets and Router Advertisement messages meant for a Mobile Node (MN) may become lost during a handoff. A Corresponding Node (CN) has to wait for a timeout before it can resend unacknowledged TCP packets. That and the duration of connection interruption

between a MN and its BS contribute to the end-to-end delay between nodes. Packet loss and degradation in transmission protocols are some of the side effects of handoffs in MIP.

MIPv6 was introduced to solve the problems which the MIPv4 faces such as the lack of multimedia transmission support and triangular routing effect. Route Optimization (RO) which was implemented in MIPv6 to eliminate the triangular routing between MN and its CN introduced other problems regarding registration and binding updates. Along with a bigger header and packet sizes a handoff in MIPv6 can be equal or greater than a MIPv4 handoff.

## **6.2 Achievements of the Study**

This study discusses the handoff procedure in MIPv6 with different implementations. The delays generated by the standard handoff in MIPv6 are too great to be neglected as they can reach up to more than three seconds. The addressing schemes are among the highest contributors to handoff latencies. Some of the addressing schemes can generate more than one second delay in the very rare duplicate address acquisition situation. Second to that come the registration processes which can generate up to 500ms even in case of successful registration. The Route Optimization which is the effective solution to MIPv4 triangular routing also introduced new kinds of delays. These are represented in the form of Corresponding Node BUs and BAs. This kind of load may even cause collisions to occur. Another type of delay generator is the movement detection technique which can also contribute up to one second delay depending on the BS and MN settings.

Many handoff schemes were proposed to solve one or more of these delay generators in MIPv6 handoff. FastRA reduced the time needed for a router to respond to router solicitations from up to 500ms to around 10ms. This was achieved by assigning one router to respond to router solicitations instantly instead of randomly. However such scheme requires manual configurations to the routers in the networks.

Advanced-DAD and Optimistic-DAD tried to solve the addressing problem by generating a pool of non duplicated addresses for a MN to choose from in case of A-DAD. On the other hand O-DAD takes advantage of low the possibility of duplicating an IP address in IPv6 to allow a MN to use an IP address without checking for duplication. While A-DAD requires manual configuration to the router it allows a MN to use DHCP to acquire an address. In the case of O-DAD the MN may actually acquire a duplicated address and cause confusion in the network. The possibility of two MNs using the same IP address can be interesting if the BS broadcasts different packets to different ports.

Optimistic Node tries to address more than one delay generator at the same time. It uses O-DAD as an addressing scheme. It requires a MN to perform a home registration with its Home Agent periodically which was proved to cause heavier signaling load than other schemes. It also predicts the success of the Home Agent registration process and starts a CN registration before ACKs from HA returns. This would cause for unnecessary signaling load in case the home registration process fails. It can also cause the packets sent from the CN to the new BS to become lost.

A new scheme (Early Handoff) was proposed, studied and compared with other schemes. In order to understand the advantages and disadvantages of the scheme a simulation model was designed. The simulation model consisted of two main scenarios with different number of nodes, movement patterns and transmission protocols.

In the first scenario, a single CN communicated with a single MN moving along a predetermine path. The second scenario simulated multiple CNs and MNs Random Waypoint Algorithm movement pattern. The single MN scenario showed that when a MN moves between BSs with overlapped coverage area, this area can be used successfully to perform an Early Handoff. The Early Handoff improved the performance of both FTP over TCP and CBR over UDP (emulating a VoIP) connections. In EH the MN does not wait until it loses RAs from its BS to assume that it is going through a handoff, thus, saving some milliseconds of packets interruption. With both transmission protocols the EH a MN experienced shorter handoffs compared to FastRA and Optimistic Node. This translated to less packets loss rate and better transmission.

In the scenarios with multiple communicating nodes the EH performed much worse than the other two schemes. That is because many MN performed unnecessary handoffs which meant more Binding Updates (BUs) and Binding Acknowledgments (BAs) messages. Such messages contributed with a higher signaling load and reduced the bandwidth for each station. They also caused more collisions than in the FastRA scheme. The Optimistic Node had a better performance than EH and worse than FastRA because it performed Home Registration process periodically, even if the MN was not moving. Higher signaling load and higher number of packet loss translated to a higher handoff delay.

This study concludes that some schemes perform better than others under certain conditions. The overlapping areas which give the advantages to EH can also be disadvantageous when used with random movement patterns.

### **6.3 Limitations**

This study faced many limitations presented in some elements that could have contributed to handoff delays were disabled. The addressing scheme in ns-2 does not actually implement IP addressing. Thus the time needed by a router to submask a packet, for example, cannot be contributed. This allows a MN to acquire a new CoA in less time than it should. Another limitation in this study is the lack of Neighbor Discovery and DHCPv6 in MobiWAN, the extension to simulate MIPv6 under (ns-2). MobiWAN itself is a very old extension and it does not follow the new MIPv6 specification document. It also works on ns-2.1b6 which is quite old and requires an old Linux in order to run. Another limitation is that the ns-2 generated traces for wired nodes only even though the simulations included wireless nodes. Therefore it was difficult to calculate the number of bytes received by the MN during TCP simulations. Using the agent (LossMonitor) helped solve that problem for UDP connections. However, this could not be used with TCP since it does not send ACKs for the received TCP packets. It was mentioned in the ns-2 web site, FAQ section that ns-2 is a software and hardware dependable program. That it produces different results on different machines depending on how the machine handles floating points operations.

### **6.4 Recommendations for Future Work**

Although this study has showed that Early Handoff is not really effective in the cellular implementations, other modifications can be made to improve its efficiency. Such modification could be to study how to implement it in satellite networks where Low Orbit Satellite travels over the Earth. The MNs cycle through a number of satellites to stay inside a coverage area.

Another possibility is to use EH as an alarming stage with other schemes where it could be used to prevent the CN from sending TCP or UDP packets. Freezing the connections between nodes can help save UDP packets from being lost and TCP from degrading due to its congestion control mechanism.

Other improvements can be done to ns-2 to support different addressing schemes and migrating MobiWAN to newer ns-2 disruptions.

## REFERENCES

- Altman, E., and Jimenez, T. (2003), “NS Simulator for Beginners”, University De Los Andes, France, Lecture Notes (2003/2004).
- Aura, T. (2005), “Cryptographically Generated Addresses (CGA)”, RFC 3972, <http://www.ietf.org/rfc/rfc3972.txt>.
- Choi, J., and Shin, D. (2002), “Fast Router Discovery with AP Notification”, <http://www.ietf.org/internet-drafts/draft-jinchoi-l2trigger-fastrd-01.txt>. (Work in Progress).
- Clark, D. (1982), “Window and Acknowledgement Strategy in TCP”, RFC 813, [http:// www.ietf.org/rfc/rfc813.txt](http://www.ietf.org/rfc/rfc813.txt).
- Daley, G., and Nelson, R. (2003), “Duplicate Address Detection Optimization using IPv6 Multicast Listener Discovery”, <http://tools.ietf.org/html/draft-daley-ipv6-mcast-dad-02>. (Work in Progress).
- Daley, G., Pentland, B., and Nelson, R. (2003), “Movement detection optimizations in mobile IPv6”. The 11th IEEE International Conference on Networks. 28 SEP. – 1 OCT. 2003. 687-692.
- Daley, G., Pentland, B., and Nelson, R. (2006), “Effects of Fast Router Advertisement on Mobile IPv6 Handovers”. Proceedings of the Eighth IEEE *International Symposium on Computers and Communication ISCC'03*. 30 June – 3 July. Kemer - Antalya, Turkey. 1, 557-562.
- Davidson, J., Peters, J., Bhatia, M., Kalidindi, S., and Mukherjee, S. (2006). Voice over IP Fundamentals (2<sup>nd</sup> ed.): CiscoPress.
- Deering, S., and Hinden, R. (1998) “Internet Protocol, Version 6 (IPv6) Specification”, RFC 2460, <http://www.ietf.org/rfc/rfc2460.txt>

- Droms, R., Bound, J., Volz, B., Lemon, T., Perkins, C., and Carney, M. (2002), “Dynamic Host Configuration Protocol for IPv6 (DHCPv6)”, <http://www.tools.ietf.org/html/draft-ietf-dhc-dhcpv6-28>.
- Egevang, K. (1994) “The IP Network Address Translator (NAT)”, RFC 1631, <http://www.ietf.org/rfc/rfc1631.txt>.
- Ernest, T. (2001), “MobiWan A NS-2.1b6 Simulation Platform for Mobile IPv6 in Wide Area Networks”, MobiWan Official Manual.
- Ernest, T. (2002), “MobiWan: ns2 extensions to study mobility in Wide-Area IPv6 Networks”. <http://www.inrialpes.fr/planete/mobiwan>.
- Fall, K., and Varadhan, K. (Eds.) (2008), “The ns Manual”.
- Fikouras, A., El Malki, K., Cvetkovic, S., and Smythe, C. (1999), “Performance of TCP and UDP during Mobile IP Handoffs in Single-Agent Subnetworks”. Wireless Communications and Networking Conference, WCNC 1999. 21-24 September. New Orleans: IEEE, 3, 1258-1262.
- Forouzan, B. (2006), TCP/IP Protocol Suite, (3<sup>rd</sup> ed.): McGraw-Hill.
- Han, Y., Choi, J., Jang, H., and Park, P. (2003), “Advance Duplicate Address Detection”, <https://datatracker.ietf.org/drafts/draft-jinchoi-l2trigger-fastrd>. (Work in Progress).
- Heidemann, J., and Huang, P. (2002), “IPAM Tutorial: Network Modeling and Traffic Analysis with ns-2”
- Johnson, D., Perkins, C., and Arkko, J. (2004), “Mobility Support in IPv6”, RFC 3775, <http://www.ietf.org/rfc/rfc3775.txt>
- Jung, J., Montgomery, D., Cheon, J., and Kahng, H. (2003), “Mobility Agent with SIP Registrar for VoIP Services”, International Conference on Human.Society@Internet No2, 18-20 June 2003, Seoul, South Korea. 2713, 454-464
- Kempf, J., Khalil, M., and Pentland, B. (2005), “IPv6 Fast Router Advertisement”, <http://tools.ietf.org/html/draft-mkhalil-ipv6-fastra-05>. (Work in Progress).
- Liu Y., Ye, M., and Zhang, H. (2003), “The Handoff Schemes in Mobile IP”. Vehicular Technology Conference, VTC 2003. 4-9 October. Orlando, Florida: The 57th IEEE Semiannual, 1, 485-489.
- McCanne, S., and Floyd, S. “ns Network Simulator”. <http://www.isi.edu/nsnam/ns/>

- Montavont, N., and Noël, T. (2002), "Handover Management for Mobile Nodes in IPv6 Networks", *Communications Magazine, IEEE*, 40(8), 38-43.
- Moore, N. (2006), "Optimistic Duplicate Address Detection (DAD) for IPv6", RFC 4429, <http://www.ietf.org/rfc/rfc4429.txt>.
- Narten, T., and Draves, R. (2001), "Privacy Extensions for Stateless Address Autoconfiguration in IPv6", RFC 3041, <http://www.ietf.org/rfc/rfc3041.txt>.
- Narten, T., Nordmark, E., and Simpson, W. (1998), "Neighbor Discovery for IPv6", RFC 2461, <http://www.ietf.org/rfc/rfc2461.txt>.
- Naugle, M. (1998), *Illustrated TCP/IP*, (1<sup>st</sup> ed): Wiley Computer Publishing, John Wiley & Sons, Inc.
- Pérez-Costa, X., Torrent-Moreno, M., and Hartenstein, H. (2003a), "A Performance Comparison of Mobile IPv6, Hierarchical Mobile IPv6, Fast Handovers for Mobile IPv6 and Their Combination". *Mobile Computing and Communications Review*, 7(3), 5-19.
- Pérez-Costa, X., Torrent-Moreno, M., and Hartenstein, H. (2003b), "A Simulation Study on the Performance of Hierarchical Mobile IPv6". *ACM SIGMOBILE Mobile Computing and Communications Review*, 7(4), 5-19.
- Perkins, C. (2002), "IP Mobility Support for IPv4", RFC 3344, <http://www.ietf.org/rfc/rfc3344.txt>.
- Postel, J. (1980), "User Datagram Protocol", RFC 768, <http://www.ietf.org/rfc/rfc768.txt>.
- Postel, J. (1981a), "Internet Protocol", RFC 791, <http://www.ietf.org/rfc/rfc791.txt>.
- Postel, J. (1981b), "Transmission Control Protocol", RFC 793, <http://www.ietf.org/rfc/rfc793.txt>.
- Stevens, W. (1997), "TCP Slow Start, Congestion Avoidance, Fast Retransmit, and Fast Recovery Algorithms", RFC 2001, <http://www.ietf.org/rfc/rfc2001.txt>.
- Thomson, S. and Narten, T. (1998), "IPv6 Stateless Address Autoconfiguration", RFC 2462, <http://www.ietf.org/rfc/rfc2462.txt>.
- Vogt, C. (2006), "A comprehensive and efficient handoff procedure for IPv6 mobility support". *Proceedings of the 2006 International Symposium on a World of Wireless, Mobile and Multimedia Networks WoWMoM'06*. 26- 29 June. Niagara Falls, Buffalo-NY: IEEE, 212-218.

- Vogt, C., and Doll, M. (2006), "Efficient End-to-End Mobility Support in IPv6".  
Wireless Communications and Networking Conference, WCNC 2006. 3-6 April.  
Las Vegas, NV: IEEE, 1, 575-580.
- Wheat, J., Hiser, R., Tucker, J., Neely, A., and Mcculloug, A. (2001), Designing a  
Wireless Network: Understand How Wireless Communication Works, (1<sup>st</sup> ed.):  
Syngress Publishing, Inc.
- Willig, A., Kubisch, M., and Wolisz, A. (2001), "Measurements and Stochastic  
Modeling of Wireless Link in an Industrial Environment", Technical University  
Berlin, Telecommunication Networks Group, TKN Technical Report  
TKN-01-001.

## APPENDIX A

### (mipv6.cc)

```

/* This software comprises contributed code made by Motorola, as a
 * Contributor, to Network Simulator NS-2 software provided by the
 * Regents of the University of California.
 * (Copyright; Regents of the University of California, 1994)
 * The contributed code was made as a result of a partnership between
 * Motorola and INRIA Rhone-Alpes.
 *
 * Copyright in the contributed code belongs to Motorola Inc. 2001
 *
 * THIS SOFTWARE IS PROVIDED BY THE REGENTS AND CONTRIBUTORS ``AS IS'' AND
 * ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
 * IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
 * ARE DISCLAIMED. IN NO EVENT SHALL THE REGENTS OR CONTRIBUTORS BE LIABLE
 * FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL
 * DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS
 * OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION)
 * HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT
 * LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY
 * OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF
 * SUCH DAMAGE.
 * ALL ADVERTISING MATERIALS MENTIONING FEATURES OR USE OF THIS SOFTWARE MUST
 * DISPLAY AN ACKNOWLEDGEMENT TO THE COPYRIGHT OWNERS.
 * ANY REDISTRIBUTION OF THIS SOFTWARE MUST CONTAIN THE ABOVE COPYRIGHT NOTICES,
 * CONDITIONS AND DISCLAIMER.
 */

/* #####
 * This code was developed by Thierry Ernst (1998-2001)
 * MOTOROLA Labs Paris FRANCE - INRIA Rhone-Alpes Grenoble (PLANETE) FRANCE
 * NS-2.1b6 enhancements for Wide-Area mobility simulations
 * #####
 */

// *****
//                               mipv6.cc
//      Mobility Support in IPv6
//      Based on <draft-ietf-mobileip-ipv6-11.txt> March 2000 with some
//      simplifications.
//
// *****

// *****
// Processing of Router Advertisement
// *****
void MNAgent::recv_ads(Packet *p)
{
    hdr_ip *iph = HDR_IP(p);
    hdr_rtads *rh = HDR_RTADS(p);

    // We search in list of BSs to check if we know about this BS

```

```

Entry *node = lookup_entry(iph->saddr(), head(bslist_head_));

Entry *node1 = head(bslist_head_); // Points to the top entry in the
list.
node1 = lookup_entry(iph->saddr(), node1->next_entry()); // Checks if the
RA is from the previous BS

if ( node ) {
    // We already have this BS in the list - Reset its expire time
    // No need to send BU

    node->update_entry(NONE, iph->saddr(), NONE, NOW, rh->lifetime());

    RA_t = NOW; // Reset RA timer.

} else if (node1 && ((NOW-RA_t) < 1.0)) {

    // The RA is from the previous BS and MN is still in overlapped
area.
    // Connection with current BS is probably still alive.
    // Do nothing!

} else {

    // New ads. This BS is not yet recorded in our BS list
    // Record it in front of list. This BS will become current BS
    // We need a new COA on the BS's subnet and we need
    // to register BS in list and new binding with HA / CN.

    if (print_info_) cout << " *** new BS ***";
    Entry *pbs = add_bs(iph->saddr());

    // XXX: I should send a request to obtain a COA
    pbs->update_entry(NONE, NONE, get_coa(iph->saddr()), NOW, rh-
>lifetime());
    coalost_ = TRUE;
    reg();
}
}

// *****
// Registration with HA, CNS, previous default router
// When ?
// => New BS
// => Timer has expired
// => BU Request (XXX: not yet implemented)
// Send current COA to:
// - HA: If we are back home, COA would be = home address. Fine.
// - Previous BS: unless previous BS is HA or we don't have one
// XXX: How long should we keep forwarding at previous BS ?
// - CNS if we have some in the Binding List
// [MIPv6] says that BUs are generally sent to CNS
// when a BU containing a new COA is sent to the HA.
// Here, we send BUs to CNS whenever we send a BU to HA.
// *****
void MNAgent::reg()
{
    // Check if we are still attached to a BS
    if ( ! head(bslist_head_) ) {
        // We do not have BS at all in the BS list
        bu_timer_.resched(max_rate_);
        send_sols();
        return;
    }

    // Check if we have an active care-of address
    if (coalost_ == TRUE) {

```

```

// We have BSs in the list, but we have lost contact with current BS
// because TIMER_BSLIST has expired.
// Choose new BS = first on list since we have deleted the former one
// and register with it

// Stop sending BUs at "previous previous" BS if still in BU List
// XXX: this should be automatic - we don't keep it very long (?)
remove_bulist(oldbs_, BU_BS);

Entry *pbs = head(bslist_head_);
oldbs_ = bs_;
bs_ = pbs->addr;
oldcoa_ = coa_;
coa_ = pbs->caddr;
adlftm_ = pbs->lifetime();

// Set the new BS as the gateway to the Internet
set_subnet();
coalost_ = FALSE;
nbu_ = 0;

// We need to establish forwarding from previous BS
// if there is one
// NOTE: if previous BS is HA, add_bulist already have
// an entry and won't add a new one
// XXX: if ( oldbs_ != NONE && oldbs_ != ha_ ) {
if ( oldbs_ != NONE ) {
    Entry *nn = add_bulist(oldbs_, BU_BS, ON);
    nn->update_entry(NONE, oldcoa_, coa_, NOW, reglftm_);
}
}

// As specified in Mobile IPv6, periodic BU should be sent
// with longer interval after 5 transmissions
if ( nbu++ < MAX_FAST_UPDATES )
    bu_timer_.resched(max_rate_);
else
    bu_timer_.resched(slow_rate_);

send_standard_bu(BU_HA);
if ( rt_opti_ ) send_standard_bu(BU_CN);
if ( bs_forwarding_ ) send_standard_bu(BU_BS);
}

```

## APPENDIX B

### (mipv6-1.tcl)

```

# Basic Mobile IPv6 example without using ns-topoman
# Needs proc defined in file proc-mipv6-config.tcl

Agent/MN set bs_forwarding_      0      ; # 1 if forwarding from previous BS
#####
proc log-mn-movement { } {
    global logtimer ns
    Class LogTimer -superclass Timer
    LogTimer instproc timeout {} {
        global mobile_
        $mobile_ log-movement
        $self sched 1
    }
    set logtimer [new LogTimer]
    $logtimer sched 1
}

#####
# Create Topology
#####
proc create-my-topo {} {
    global ns opt topo mobile_ cn_

    # Create and define topography
    set topo [new Topography]
    $topo load_flatgrid 500 1500

    # god is a necessary object when wireless is used
    create-god 4

    # Call node-config
    $ns node-config \
        -addressType hierarchical

    # Set NS Addressing
    AddrParams set domain_num_ 2
    AddrParams set cluster_num_ {1 4}
    AddrParams set nodes_num_ {1 1 2 1 1}

#####
    # Create Nodes
#####

    # Create Correspoanet Node
    set cn_ [create-host 0.0.0]

    # Create Central Router

```

```

set router_ [create-router 1.0.0]

# Create Base Stations
set bs1_ [create-base-station 1.1.0 1.0.0 100 300 0]
set bs2_ [create-base-station 1.2.0 1.0.0 100 750 0]
set bs3_ [create-base-station 1.3.0 1.0.0 100 1200 0]

# Create Mobile Node
set mobile_ [create-mobile 1.1.1 1.1.0 100 300 0 0 0]

# Create Links between the Central Router and the Base Stations
$ns duplex-link $cn_ $router_ 100Mb 10.80ms DropTail
$ns duplex-link $router_ $bs1_ 100Mb 1.80ms DropTail
$ns duplex-link $router_ $bs2_ 100Mb 1.80ms DropTail
$ns duplex-link $router_ $bs3_ 100Mb 1.80ms DropTail

display_ns_addr_domain
}

#####
# End of Simulation
#####
proc finish { } {
    global tracef ns namf opt mobile_ cn_

    # Dump the Binding Update List of MN and Binding Cache of HA
    [$mobile_ set ha_] set regagent_ dump
    [$cn_ set regagent_] dump
    [$mobile_ set regagent_] dump

    $ns flush-trace
    flush $tracef
    close $tracef
    exit 0
}

#####
# Main
#####
proc main { } {
    global opt ns namf n tracef mobile_ cn_ sink tracebw traceps

    # Source Files
    source ../ns-orig-src/tcl/mobility/timer.tcl

    #>----- Extract options from command line -----<
    Getopt      ; # Get option from the command line
    DisplayCommandLine ; #Show the command line
    #>----- Simulator Settings -----<
    set ns [new Simulator]

    #>----- Open trace files -----<
    set tracef [open $opt(tracefile) w]
    set tracebw [open bw-1.tr w]
    set traceps [open pl-1.tr w]

    #... dump the file
    $ns trace-all $tracef

    #>----- Protocol and Topology Settings -----<
    create-my-topo
    log-mn-movement

    set-cbr
    $ns at 0.01 "record"

    #Move the mobile node

```

```

$ns at 10.0 "$mobile_ setdest 100 1200 10"
$ns at 200.0 "$mobile_ setdest 100 300 10"
$ns at 400.0 "$mobile_ setdest 100 1200 10"

#>----- Run Simulation -----<

$ns at $opt(stop) "finish"
$ns run
}

#####
# Create CBR over UDP traffic
#####
proc set-cbr { } {
    global ns cn_ mobile_ sink
    set udp [new Agent/UDP]
    $ns attach-agent $cn_ $udp
    set sink [new Agent/LossMonitor]
    $ns attach-agent $mobile_ $sink
    $ns connect $udp $sink

    set cbr [new Application/Traffic/CBR]
    $cbr attach-agent $udp
    $cbr set packetSize_ 200
    $cbr set rate_ 64Kb

    $ns at 1.0 "$cbr start"
}

#####
# Create FTP over TCP traffic
#####
proc set-ftp { } {
    global ns cn_ mobile_ sink
    set tcp [new Agent/TCP]
    $ns attach-agent $cn_ $tcp
    set sink [new Agent/TCPSink]
    $ns attach-agent $mobile_ $sink
    $ns connect $tcp $sink

    set ftp [new Application/FTP]
    $ftp attach-agent $tcp

    $ns at 1.0 "$ftp start"
}

#####
# Create Bandwidth per Second and PacketLoss in UDP trace files
#####
proc record {} {
    global tracebw sink traceps
    set ns [Simulator instance]
    set time 1.0 ; #Set time to 1 s for the Bandwidth per Second measures
    #set time 0.01 ; #Set time to 10ms for Packet Loss
    set bw [$sink set bytes_]
    #set pseq [$sink set seqno_]
    set now [$ns now]
    puts $tracebw "$now [expr $bw*8/1024] $pseq"
    #puts $traceps "$now $pseq"
    $sink set bytes_ 0.0
    $ns at [expr $now+$time] "record"
}

main

```

## APPENDIX C

### (mipv6-30.tcl)

```

# Basic Mobile IPv6 example without using ns-topoman
# Needs proc defined in file proc-mipv6-config.tcl

Agent/MN set bs_forwarding_      0      ; # 1 if forwarding from previous BS
#####
# End of Simulation
#####
proc finish { } {
    global tracef ns namf opt

    # Dump the Binding Update List of MN and Binding Cache of HA
    [[$mobile_ set ha_] set regagent_] dump
    [$cn_ set regagent_] dump
    [$mobile_ set regagent_] dump

    $ns flush-trace
    flush $tracef
    close $tracef
    exit 0
}

#####
# Main
#####
proc main { } {
    global opt ns namf n tracef sink tracebw traceps topo

    # Source Files
    source ../ns-orig-src/tcl/mobility/timer.tcl

    #>----- Extract options from command line -----<
    Getopt      ; # Get option from the command line
    DisplayCommandLine

    #>----- Simulator Settings -----<
    set ns [new Simulator]

    #>----- Open trace files -----<

    set tracef [open $opt(tracefile) w]
    set tracebw [open bw-30.tr w]
    set traceps [open pl-30.tr w]
    #... dump the file
    $ns trace-all $tracef

```

```

#>----- Protocol and Topology Settings -----<
# Create and define topography
set topo [new Topography]
$topo load_flatgrid 500 1500

# god is a necessary object when wireless is used
create-god 33

# Call node-config
$ns node-config \
    -addressType hierarchical

# Set NS Addressing
AddrParams set domain_num_ 2
AddrParams set cluster_num_ {1 4}
AddrParams set nodes_num_ {30 1 10 10 10}

set Ran [new RNG]
$Ran seed 1

#####
# Create Nodes
#####

# Create Correspoanet Nodes
for {set i 1} {$i < 101} {incr i} {
    set cn_($i) [create-host 0.0.$i]
}

# Create Central Router
set router_ [create-router 1.0.0]

# Create Base Stations
set bs1_ [create-base-station 1.1.0 1.0.0 100 300 0]
set bs2_ [create-base-station 1.2.0 1.0.0 100 750 0]
set bs3_ [create-base-station 1.3.0 1.0.0 100 1200 0]

# Create Mobile Node
for {set i 1} {$i < 11} {incr i} {
    set x1 [new RandomVariable/Uniform]
    $x1 use-rng $Ran
    $x1 set min_ 10.0
    $x1 set max_ 540.0
    set mobile_($i) [create-mobile 1.1.$i 1.1.0 100 $x1 0 1 1]
}
for {set i 11} {$i < 21} {incr i} {
    set x2 [new RandomVariable/Uniform]
    $x2 use-rng $Ran
    $x2 set min_ 570.0
    $x2 set max_ 990.0
    set mobile_($i) [create-mobile 1.2.$i 1.2.0 100 $x2 0 1 10]
}

for {set i 21} {$i < 31} {incr i} {
    set x3 [new RandomVariable/Uniform]
    $x3 use-rng $Ran
    $x3 set min_ 1050.0
    $x3 set max_ 1450.0
    set mobile_($i) [create-mobile 1.3.$i 1.3.0 100 $x3 0 1 15]
}

# Create Links between the Central Router and the Base Stations
for {set i 1} {$i < 31} {incr i} {
    $ns duplex-link $cn_($i) $router_ 100Mb 10.80ms DropTail
}

```

```

}

$ns duplex-link $router_ $bs1_ 100Mb 1.80ms DropTail
$ns duplex-link $router_ $bs2_ 100Mb 1.80ms DropTail
$ns duplex-link $router_ $bs3_ 100Mb 1.80ms DropTail

display_ns_addr_domain
#####
# Create CBR over UDP traffic with LossMonitor for a single MN
#####
set udp [new Agent/UDP]
$ns attach-agent $cn_(1) $udp
set sink [new Agent/LossMonitor]
$ns attach-agent $mobile_(1) $sink
$ns connect $udp $sink

set cbr [new Application/Traffic/CBR]
$cbr attach-agent $udp
$cbr set packetSize_ 200
$cbr set rate_ 64Kb

$ns at 0.01 "record"
$ns at 1.0 "$cbr start"

#####
# Create CBR over UDP traffic for 29 MNs
#####
for {set i 2} {$i < 31} {incr i} {
    set x [new RandomVariable/Uniform]
    $x use-rng $Ran
    $x set max_ 50.0
    $x set min_ 1.0

    set udp_($i) [new Agent/UDP]
    $ns attach-agent $cn_($i) $udp_($i)
    set null_($i) [new Agent/Null]
    $ns attach-agent $mobile_($i) $null_($i)
    $ns connect $udp_($i) $null_($i)

    set cbr_($i) [new Application/Traffic/CBR]
    $cbr_($i) attach-agent $udp_($i)
    $cbr_($i) set packetSize_ 200
    $cbr_($i) set rate_ 64Kb

    $ns at $x "$cbr_($i) start"
}

#>----- Run Simulation -----<
$ns at $opt(stop) "finish"
$ns run
}

#####
# Create Bandwidth per Second and PacketLoss in UDP trace files
#####
proc record {} {
    global tracebw sink traceps
    set ns [Simulator instance]
    set time 1.0 ; #Set time to 1 s for the Bandwidth per Second measures
    #set time 0.01 ; #Set time to 10ms for Packet Loss
    set bw [$sink set bytes_]
    #set pseq [$sink set seqno_]
    set now [$ns now]
    puts $tracebw "$now [expr $bw*8/1024]"
    #puts $traceps "$now $pseq"
    $sink set bytes_ 0.0
    $ns at [expr $now+$time] "record"
}
main

```

**APPENDIX D****(mipv6-1.sh)**

```
#!/bin/tcsh

# #####
# Demonstration of Mobile IPv6 on a very simple network
# no need for TOPOMAN
# #####
set NS=mobiwan          # NS binary

# Output Directory
set OUTDIR=mipv6

set NAME=${OUTDIR}/out.nam      # NAM file (optional)
set TRACEF=${OUTDIR}/out-1.tr
set INFOF=${OUTDIR}/out.info

# Launch simulation
$NS mipv6.tcl -mactrace OFF -NAM 0 -stop 600 -tracefile $TRACEF
-----
```

**APPENDIX E****(mipv6-30.sh)**

```
#!/bin/tcsh

# #####
# Demonstration of Mobile IPv6 on a very simple network
# no need for TOPOMAN
# #####
  set NS=mobiwan          # NS binary

# Output Directory
set OUTDIR=mipv6

set NAMF=${OUTDIR}/out.nam      # NAM file (optional)
set TRACEF=${OUTDIR}/out-30.tr
set INFOF=${OUTDIR}/out.info

# Launch simulation
$NS mipv6-100.tcl -mactrace OFF -NAM 0 -stop 600 -tracefile $TRACEF
```