

**AN OPTIMIZED FRAMEWORK FOR HEADER  
SUPPRESSION OF REAL TIME IPv6 TRAFFIC IN  
MULTIPROTOCOL LABEL SWITCHING (MPLS)  
NETWORKS**

**By**

**IMAD JASIM MOHAMMED**

**Thesis submitted in fulfillment of the requirements**

**for the degree of**

**Doctor of Philosophy**

**August 2011**

## ACKNOWLEDGEMENTS

بِسْمِ اللَّهِ الرَّحْمَنِ الرَّحِيمِ  
{نرفع درجات من نشاء وفوق كل ذي علم عليم "76"} (سورة يوسف)

I would like to take this opportunity to convey my sincere thanks and deepest gratitude to my supervisor, **Dr. Wan Tat Chee**, for all the help and valuable guidance provided to me throughout my period of research. I consider myself privileged to have had the opportunity to work under his guidance. I'm also grateful to my co-supervisor **Dr. Putra Sumari** for his help and support.

Moreover, I would like to convey my appreciation to **Prof. Sures** (NAv6 Director), **Dr. Andrew Meulenberg** and to **Yung-Wey Chong** (NAv6), all NAv6 center members, the School of Computer Sciences, the Institute of Postgraduate Studies, and the university library for their help and support.

My sincere gratitude also goes to **my parents**, my wonderful wife **Hind**, **brothers**, **sisters**, and our children **Noor**, **Mohammed**, **Hajer**, and **Yaseen**. I thank them for their support, understanding, and encouragement during every step of my study and writing of this thesis.

The favour, above all, before all, and after all, is entirely Allah's, to whom my never-ending thanks and praise are humbly due.

Thank you!

Imad Jasim Mohammed  
Penang, Malaysia, March 2011

## **DEDICATION**

This thesis is dedicated to my mother, father, wife, brothers, sisters and children for their patience and the encouragement they provided during the entire period of the study.

## TABLE OF CONTENTS

	<b>Page</b>
<b>ACKNOWLEDGEMENTS</b> .....	<b>ii</b>
<b>DEDICATION</b> .....	<b>iii</b>
<b>TABLE OF CONTENTS</b> .....	<b>iv</b>
<b>LIST OF TABLES</b> .....	<b>xi</b>
<b>LIST OF FIGURES</b> .....	<b>xiv</b>
<b>LIST OF ABBREVIATIONS</b> .....	<b>xxii</b>
<b>ABSTRAK</b> .....	<b>xxv</b>
<b>ABSTRACT</b> .....	<b>xxvii</b>
<b>CHAPTER ONE: INTRODUCTION</b> .....	<b>1</b>
1.1 Background.....	1
1.2 Problem Statement.....	3
1.3 Research Motivation .....	4
1.4 Objective .....	4
1.5 Scope .....	5
1.6 Research Framework .....	6
1.7 Thesis Organization .....	7
<b>CHAPTER TWO: LITERATURE REVIEW</b> .....	<b>9</b>
2.1 Introduction .....	9
2.2 QoS Models, Mechanisms and Metrics .....	10
2.2.1 Network QoS Requirements for VoIP Applications .....	11
2.2.1.1 Delay .....	11
2.2.1.2 Jitter .....	12
2.2.1.3 Packet Drop .....	12

2.2.1.4	Throughput .....	13
2.2.2	QoS Models .....	13
2.2.2.1	Best Effort (No QoS) .....	13
2.2.2.2	Integrated Services (Hard QoS) .....	14
2.2.2.3	Differentiated Services (Soft QoS) .....	14
2.2.3	Router Operational Planes .....	14
2.2.3.1	Data Plane .....	15
2.2.3.2	Control Plane .....	15
2.3	IP Protocols for Internet .....	16
2.4	Multiprotocol Label Switched (MPLS) Technology .....	17
2.4.1	MPLS Domain .....	19
2.4.2	MPLS Packet Processing and Forwarding Mechanism .....	20
2.4.3	MPLS Header Structure .....	21
2.4.4	MPLS Support for Real Time Applications .....	22
2.4.5	MPLS Enabled Backbone .....	22
2.4.6	MPLS Signaling Protocols .....	24
2.4.7	Explicit Label Switched Path (Explicit LSP) .....	24
2.4.8	DS-TE MPLS Fusion .....	25
2.5	Improvement of Response Time (QoS parameter) .....	27
2.6	IPv6 Transition Mechanisms .....	28
2.7	Header Compression and Suppression Schemes .....	30
2.7.1	General Header Compression Development Challenges.....	32
2.7.1.1	Storage Requirements .....	32
2.7.1.2	Processing Requirements .....	32
2.7.1.3	Restrictions from Underlying Layers .....	33

2.7.1.4	Security Consideration .....	33
2.7.2	Payload Header Suppression (PHS) .....	33
2.7.3	RObust Header Compression (ROHC) .....	35
2.7.4	PHS verses ROHC (Functional Analysis) .....	36
2.7.4.1	CRC Dependency .....	41
2.7.4.2	Field Encoding and Decoding .....	41
2.7.4.3	Space Complexity .....	41
2.7.4.4	Unidirectional Property of MPLS-LSP .....	42
2.7.4.5	Size of Static and Dynamic Fields in Profiles .....	42
2.7.4.6	Load, Cost, and Properties of Transmission Domain .....	43
2.7.4.7	Simple Numerical Analysis .....	44
2.8	Header Compression Schemes over MPLS .....	44
2.9	Chapter Summary .....	46
	<b>CHAPTER THREE: The Proposed MPHS Framework for MPLS .....</b>	<b>48</b>
3.1	Introduction .....	48
3.2	Overview for MPHS Framework .....	48
3.2.1	Justifications for MPHS .....	52
3.2.2	Requirements for MPHS Framework .....	53
3.2.3	Analysis of MPHS verses PHS and ROHC using FSMs .....	54
3.3	Design of MPHS-based Network using Basic MPLS Topology .....	57
3.3.1	Headers and Protocols under MPHS Support .....	60
3.3.2	MPHS Headers Format .....	61
3.3.2.1	MPHS-N Header Format for Native IPv6 Support .....	61
3.3.2.2	MPHS-T Header Format for 6-in-4 Support .....	63
3.3.2.3	Suppression Gain using MPHS-N and MPHS-T .....	64

3.4	MPHS Algorithms .....	64
3.4.1	Context Identifier (CID) Generator Algorithm .....	64
3.4.2	MPHS SUPPRESSOR Algorithm .....	65
3.4.2.1	Configuration for MPHS SUPPRESSOR .....	66
3.4.2.2	Tasks of MPHS SUPPRESSOR .....	66
3.4.2.3	MPHS SUPPRESSOR in Ingress (Structure) .....	67
3.4.2.4	Flowchart for MPHS SUPPRESSOR .....	68
3.4.3	MPHS RESTORER Algorithm .....	72
3.4.3.1	Configuration for MPHS RESTORER .....	72
3.4.3.2	Tasks of MPHS RESTORER .....	72
3.4.3.3	MPHS RESTORER in Penultimate .....	73
3.4.3.4	Flowchart for MPHS RESTORER .....	73
3.4.4	Extension of LDP Algorithms .....	75
3.4.5	CID Interpreter Algorithm .....	77
3.5	MPHS and Explicit LSP Support .....	78
3.6	Message Passing for MPHS .....	79
3.7	Chapter Summary .....	83
	<b>CHAPTER FOUR: MPHS MODELING AND VALIDATION .....</b>	<b>84</b>
4.1	Introduction .....	84
4.2	Analytical model for MPHS .....	84
4.2.1	Model Architecture .....	84
4.2.2	Model Parameters, Notation, and Delay Calculations .....	87
4.2.3	Delay Components in MPHS .....	89
4.3	Model Performance Validation and Simulation .....	92
4.3.1	MPHS Topology and Parameters .....	92

4.3.2	Model Calculations and Validation .....	94
4.3.2.1	Estimated Upper-Bound Delay using MPHS .....	94
4.3.2.2	Estimated Queuing Delay, Throughput and Drop using MPHS .....	95
4.3.2.3	End-to-End Delay Calculations .....	97
4.3.2.4	Queue Buffer Utilization .....	100
4.4	Chapter Summary .....	100
<b>CHAPTER FIVE: SIMULATION ENVIRONMENT FOR MPHS .....</b>		<b>101</b>
5.1	Introduction .....	101
5.2	Overview of Network Simulator version 2 (NS2) .....	101
5.3	IP Queuing in NS2 .....	101
5.4	MPHS and NS2 Routing Modules .....	102
5.5	Label Distribution Modes for MPLS in NS2 .....	104
5.6	MPHS Design and Integration in NS2 .....	104
5.6.1	MPHS Node Architecture .....	105
5.6.2	MPHS SUPPRESSOR .....	107
5.6.3	MPHS RESTORER .....	107
5.6.4	MPHS Classifier and Switching Tables .....	108
5.7	Overview of Thesis Experiments Using NS2 .....	109
5.8	Detailed Simulation Parameters .....	111
5.8.1	Throughput .....	111
5.8.2	One-way Delay .....	112
5.8.3	UDP Packets Drop or TCP Retransmission .....	112
5.8.4	Jitter (delay variation) .....	112
5.8.5	Response Time for Web Traffic .....	113



5.9	Chapter Summary .....	113
	<b>CHAPTER SIX: RESULTS, ANALYSIS AND DISCUSSIONS .....</b>	<b>114</b>
6.1	Introduction .....	114
6.2	<b>Experiment 1: MPHS Performance Evaluation using Basic MPLS</b>	<b>115</b>
	Topology	
6.2.1	Topology and Simulation Parameters .....	113
6.2.2	Analytical Results .....	116
6.2.3	Performance Evaluation and Discussion .....	117
6.2.4	Outcome of Experiment 1 .....	121
6.3	<b>Experiment 2: MPHS Performance Evaluation using Complex</b>	<b>125</b>
	MPLS Topology	
6.3.1	Topology and Simulation Parameters .....	125
6.3.2	Analytical Results .....	128
6.3.3	Performance Evaluation and Discussion .....	131
	6.3.3.1 LSP0 and LSP2 (Normal and shared LSPs) .....	131
	6.3.3.2 Explicit LSP (LSP1) .....	142
6.3.4	Outcome of Experiment 2 .....	144
6.4	<b>Experiment 3: MPHS Performance Evaluation using</b>	<b>150</b>
	Heterogeneous Traffic	
6.4.1	Topology and Simulation Parameters .....	150
6.4.2	Analytical Results .....	152
	6.4.2.1 Expected Available Link-Capacity for TCP	153
	(TCP Analytical Results)	
	6.4.2.2 UDP Analytical Results .....	153
6.4.3	Performance Evaluation and Discussion .....	155

6.4.4	Outcome of Experiment 3 .....	157
6.5	<b>Experiment 4: MPHS Performance Evaluation using Web Traffic</b>	163
6.5.1	Topology and Simulation Parameters .....	163
6.5.2	Performance Evaluation and Discussion .....	166
6.5.3	Outcome of Experiment 4 .....	167
6.6	Chapter Summary .....	175
	<b>CHAPTER SEVEN: CONCLUSION AND FUTURE WORK .....</b>	<b>177</b>
7.1	Summary of Research Findings .....	177
7.1.1	Research Data .....	177
7.1.2	Research Contribution .....	178
7.2	Future Work	180
	<b>REFERENCES .....</b>	<b>181</b>
	<b>APPENDICES .....</b>	<b>188</b>
	<b>APPENDIX A: OVERVIEW FOR MPLS IN NS2 .....</b>	<b>189</b>
A.1	Overview for Label Distribution Modes for MPLS in NS2 .....	189
A.2	MPLS Node in NS2 .....	191
	<b>APPENDIX B: MPHS CODING DESCRIPTION .....</b>	<b>195</b>
	<b>LIST OF PUBLICATIONS .....</b>	<b>196</b>

## LIST OF TABLES

		<b>Page</b>
Table 2.1	ITU G.114 Determination of the Effects of Absolute Delay by the E-model	12
Table 2.2	IPv6 over MPLS Deployment Scenario	18
Table 2.3	DSCP to Service Class Mapping	23
Table 3.1	Qualitative Functional Comparison for MPHS verses ROHC and PHS	56
Table 3.2	Suppression Gain using MPHS	64
Table 4.1	Expected Queuing for D/D/1 Model	89
Table 4.2	Parameters of Topology (Figure 4.2)	93
Table 4.3	Service Time and Delay Boundary	95
Table 4.4	Model Calculations with and without MPHS	96
Table 4.5	Analytical Calculations for End-to-End Delay using “without” Scenario	98
Table 4.6	Analytical Calculations for End-to-End Delay using “native” Scenario	98
Table 6.1	Simulation Parameters of Topology	116

	( Experiment 1)	
Table 6.2	Bandwidth Saving (Experimen1 Topology)	119
Table 6.3	Simulation Parameters of Topology	126
	(Experiment 2)	
Table 6.4	Analytical Calculations for End-to-End Delay using “without” Scenario (LSP0-Experiment 2)	128
Table 6.5	Analytical Calculations for End-to-End delay using “native” Scenario (LSP0-Experiment 2)	128
Table 6.6	Analytical Calculations for End-to-End Delay using “without” Scenario (LSP2-Experiment 2)	129
Table 6.7	Analytical Calculations for End-to-End Delay using “native” Scenario (LSP2-Experiment 2)	129
Table 6.8	Analytical Calculations for End-to-End Delay using “without” Scenario (LSP1-Experiment 2)	130
Table 6.9	Analytical Calculations for End-to-End Delay using “native” Scenario (LSP1-Experiment 2)	130
Table 6.10	Bandwidth Saving in Shared Path for LSP0 and LSP2 (Experiment 2)	132
Table 6.11	Bandwidth Saving in non Shared Links of LSP0 and LSP2 (Experiment 2)	132

Table 6.12	Bandwidth Saving in Explicit-LSP (LSP1)	142
Table 6.13	Summary for Results of Experiment 2	145
Table 6.14	Simulation Parameters of Experiment 3	151
Table 6.15	Expected Available Link-Capacity for TCP using “without” MPHS Scenario for UDP	153
Table 6.16	Expected Available Link Capacity for TCP using “native” MPHS Scenario for UDP	153
Table 6.17	Analytical Calculations for End-to-End Delay using “without” Scenario	154
Table 6.18	Analytical Calculations for End-to-End Delay using “native” Scenario	154
Table 6.19	Simulation Parameters of Experiment 4	164
Table 6.20	Results Summary for the Four Experiments	176

## LIST OF FIGURES

		Page
Figure 1.1	Thesis Scope	5
Figure 1.2	Research Framework	6
Figure 2.1	Thesis Interest & Boundary	9
Figure 2.2	QoS Models	13
Figure 2.3	IPv4 Header Format	16
Figure 2.4	IPv6 Header Format	17
Figure 2.5	MPLS Domain	19
Figure 2.6	MPLS Header Structure	21
Figure 2.7.	Generic IPv6-in-IPv4 Encapsulation	29
Figure 2.8	IPv6 Packet Transmission	34
Figure 2.9	Modes of Operation in ROHC	36
Figure 2.10	Size of Static and Dynamic Fields in Three Packet Header Profiles	43
Figure 3.1	MPHS Algorithms	49
Figure 3.2	Block Diagram for MPHs Framework	51
Figure 3.3	PHS, MPHs and ROHC Finite State Machines and Functional Relationship	55

Figure 3.4	A Basic MPLS Topology Served by MPHS-N and MPHS-T	58
Figure 3.5	The End-to-End Design of MPHS for LSP	59
Figure 3.6	6-in-4-in-MPLS Encapsulation	60
Figure 3.7	IPv6 Header Format under MPHS Service	60
Figure 3.8	IPv4 Header Format under MPHS Service	60
Figure 3.9	MPHS-N Format	63
Figure 3.10	MPHS-T Header Format	63
Figure 3.11	Ingress Structure with and without MPHS SUPPRESSOR	68
Figure 3.12	MPHS SUPPRESSOR in Ingress LSR (Flowchart)	69
Figure 3.13	Penultimate Structure with and without MPHS RESTORER	73
Figure 3.14	MPHS RESTORER in Penultimate (Flowchart)	75
Figure 3.15	An MPLS Topology that Supports Explicit LSP	79
Figure 3.16	Message Passing Between MPHS Objects	82
Figure 4.1	End-to-End Delay Model using MPHS	86
Figure 4.2	MPHS Topology	93
Figure 4.3	End-to-End Delay –Analytical and Simulation Results	99

Figure 4.4	Throughput - Analytical and Simulation results	99
Figure 5.1	Interaction among Node, Routing Module, and Routing.	103
Figure 5.2	MPHS and Label Distribution Modes in NS2	104
Figure 5.3	MPHS in MPLS Node Architecture (using NS2)	106
Figure 5.4	Simple Structure & Behavior for MPLS Classifier and Switching Tables in NS2 (with MPHS)	108
Figure 5.5	Experiments 1 & 2 (Real Time IPv6 & 6-in-4 Traffic)	110
Figure 5.6	Experiments 3 & 4 (Heterogeneous IPv6 Traffic)	110
Figure 6.1	<b>Experiment 1 (Topology)</b>	115
Figure 6.2	Exp.1: Throughput Per-Flow (Kbps)	122
Figure 6.3	Exp.1: Throughput Average (Kbps)  (Analytical and Simulation results)	122
Figure 6.4	Exp.1: Throughput Variance (Kbps)	122
Figure 6.5	Exp.1: Drop Average (Kbps)	122
Figure 6.6	Exp.1: Delay Per-Flow (ms)	123
Figure 6.7	Exp.1: Delay average (Analytical and Simulation Results in ms)	123
Figure 6.8	Exp.1: Delay Variance (ms)	123



Figure 6.9	Exp.1: Jitter Per Flow (ms)	124
Figure 6.10	Exp.1: Jitter Average (ms)	124
Figure 6.11	Exp.1: Jitter Variance (ms)	124
Figure 6.12	<b>Experiment 2 (Topology)</b>	127
Figure 6.13	Exp.2: Throughput Per-Flow (Kbps) - LSP0	134
Figure 6.14	Exp.2: Throughput Average (Kbps) – LSP0  (Analytical and Simulation results)	134
Figure 6.15	Exp.2: Throughput Variance (Kbps) – LSP0	134
Figure 6.16	Exp.2: Drop Average (Kbps) – LSP0	134
Figure 6.17	Exp.2: Delay Per-Flow (ms) – LSP0	135
Figure 6.18	Exp.2: Delay Average (Analytical and Simulation results in ms) -LSP0	135
Figure 6.19	Exp.2: Delay Variance (ms) – LSP0	135
Figure 6.20	Exp.2: Jitter Per-Flow (ms) – LSP0	136
Figure 6.21	Exp.2: Jitter Average (ms) – LSP0	136
Figure 6.22	Exp.2: Jitter Variance (ms) – LSP0	136
Figure 6.23	Exp.2: Throughput Per-Flow (Kbps) – LSP2	139
Figure 6.24	Exp.2: Throughput Average (Kbps) – LSP2	139

(Analytical and Simulation results)

Figure 6.25	Exp.2: Throughput Variance (Kbps) – LSP2	139
Figure 6.26	Exp.2: Drop Average (Kbps) – LSP2	139
Figure 6.27	Exp.2: Delay Per-Flow (ms) – LSP2	140
Figure 6.28	Exp.2: Delay Average (Analytical and Simulation results in ms) – LSP2	140
Figure 6.29	Exp.2: Delay Variance (ms) – LSP2	140
Figure 6.30	Exp.2: Jitter Per-Flow (ms) – LSP2	141
Figure 6.31	Exp.2: Jitter Average (ms) – LSP2	141
Figure 6.32	Exp.2: Jitter Variance (ms) – LSP2	141
Figure 6.33	Exp.2: Throughput Per-Flow (Kbps) – LSP1	147
Figure 6.34	Exp.2: Throughput Average (Kbps) – LSP1	147
	(Analytical and Simulation results)	
Figure 6.35	Exp.2: Throughput Variance (Kbps) – LSP1	147
Figure 6.36	Exp.2: Drop Average (Kbps) – LSP1	147
Figure 6.37	Exp.2: Delay Per-Flow (ms) – LSP1	148
Figure 6.38	Exp.2: Delay Average (Analytical and Simulation results in ms) – LSP1	148
Figure 6.39	Exp.2: Delay Variance (ms) – LSP1	148

Figure 6.40	Exp.2: Jitter Per-Flow (ms) – LSP1	149
Figure 6.41	Exp.2: Jitter Average (ms) – LSP1	149
Figure 6.42	Exp.2: Jitter Variance (ms) – LSP1	149
Figure 6.43	<b>Experiment 3 (Topology)</b>	152
Figure 6.44	Exp.3: Statistics for UDP-Drop and TCP-Retransmit (with and without MPHS)	158
Figure 6.45	Exp.3: TCP0 and TCP1 Per-Flow Throughput “without” MPHS	158
Figure 6.46	Exp.3: TCP0 and TCP1 Per-Flow Throughput “with” MPHS	158
Figure 6.47	Exp.3: Aggregated TCP throughput (Analytical and Simulation results for “without” and “with” MPHS)	159
Figure 6.48	Exp.3: Aggregated TCP retransmission (dropped data) for “without” and “with” MPHS	159
Figure 6.49	Exp.3: Per-Flow UDP Throughput “without” MPHS	160
Figure 6.50	Exp.3: Per-Flow UDP Throughput “native” MPHS	160
Figure 6.51	Exp.3: UDP Throughput Avg (Kbps)	160
Figure 6.52	Exp.3: UDP Throughput Variance (Kbps)	160
Figure 6.53	Exp.3: UDP Drop Avg (Kbps)	160
Figure 6.54	Exp.3: Per-Flow UDP Delay “without” MPHS	161
Figure 6.55	Exp.3: Per-Flow UDP Delay “native” MPHS	161

Figure 6.56	Exp.3: UDP Delay average	161
Figure 6.57	Exp.3: UDP Delay Variance	161
Figure 6.58	Exp.3: Per-Flow UDP Jitter “without” MPHS	162
Figure 6.59	Exp.3: Per-Flow UDP Jitter “native” MPHS	162
Figure 6.60	Exp.3: UDP Jitter Average	162
Figure 6.61	Exp.3: UDP Jitter Variance	162
Figure 6.62	<b>Experiment 4 (Topology)</b>	165
Figure 6.63	Exp.4: Statistics for UDP-Drop and TCP-Retransmit(with and without MPHS)	169
Figure 6.64	Exp.4: TCP1 and TCP2 Per-flow Throughput “without” MPHS	169
Figure 6.65	Exp.4: TCP1 and TCP2 per-flow throughput “with” MPHS	169
Figure 6.66	Exp.4: Response time for Client-Server Web-traffic (with and without MPHS)	170
Figure 6.67	Exp.4: Aggregated TCP throughput (Analytical and Simulation results for “with” and “without” MPHS)	171
Figure 6.68	Exp.4: Aggregated TCP retransmission (dropped data) for “with” and “without” MPHS	171
Figure 6.69	Exp.4: Per-flow UDP Throughput “without” MPHS	172

Figure 6.70	Exp.4: Per-flow UDP Throughput “native” MPHS	172
Figure 6.71	Exp.4: UDP Throughput avg (Kbps)	172
Figure 6.72	Exp.4: UDP Throughput variance (Kbps)	172
Figure 6.73	Exp.4: UDP Drop Avg (Kbps)	172
Figure 6.74	Exp.4: Per-Flow UDP Delay “without” MPHS	173
Figure 6.75	Exp.4: Per-Flow UDP Delay “with” MPHS	173
Figure 6.76	Exp.4: UDP Delay Average	173
Figure 6.77	Exp.4: UDP Delay Variance	173
Figure 6.78	Exp.4: Per-Flow UDP Jitter “without” MPHS	174
Figure 6.79	Exp.4: Per-Flow UDP Jitter “native” MPHS	174
Figure 6.80	Exp.4: UDP Jitter Average	174
Figure 6.81	Exp.4: UDP Jitter Variance	174

## LIST OF ABBREVIATIONS

ARPANET	Advanced Research Project Agency Network
ATM	Asynchronous Transfer Mode
BER	Bit Error Rate
BS	Base Station
CE	Customer Edge
CID	Context Identifier
CoS	Class of Service
CRC	Cyclical Redundancy Checking:
CR-LDP	Constrained-based Routing - Label Distribution Protocol
CSPF	Shortest Path First algorithm
CT	Class Type
DestIP	Destination IP
DRR	Deficit Round Robin
DSCP	Differentiated Service Code Point
ECRTP	Enhanced Compressed RTP
ERT	Explicit Routing Table
EXP	Experimental
FEC	Forwarding Equivalent Class
FO	First Order
FSM	Finite State Machine
FTP	File Transfer Protocol
GSM	Global System for Mobile Communications
HC	Header Compression
HD	Header Decompression
HTTP	Hyper Text Transfer Protocol
IETF	Internet Engineering Task Force

IGP	Internet Gateway Protocol
IPHC	Internet Protocol Header Compression
IPv6	Internet Protocol version 6
IR	Initialization and Refresh
ISP	Internet Service Provider
ITU-T	International Telecommunication Union - Telecommunication Standardization Sector
LAN	Local Area Network
LDP	Label Distribution Protocol
LER	Label Edge Router
LIB	Label Information Base
LSP	Label Switched Path
LSR	Label Switching Router
Mbone	Multicast Backbone (Internet)
MPHS	Multiprotocol Payload Header Suppression
MPLS	Multiprotocol Label Switching
native	IPv6 with MPHS
NS2	Network Simulator version 2
OSI	Open System Interconnections
PE	Provider Edge
PFT	Partial Forwarding Table
PHS	Payload Header Suppression
PHSF	Payload Header Suppression Field
PPP	Point-to-Point Protocol
PPS	Packet Per Second
PW	Pseudo Wire
QoS	Quality of Service
RESV	Reservation
ROHC	RObust Header Compression

RSVP	Resource Reservation Protocol
RTP	Real-Time Transport Protocol
SIP	Session Initiation Protocol
SLA	Service Level Agreement
SO	Second Order
SS	Subscriber Station
TCP	Transmission Control Protocol
TCP	Traffic Conditioning
TE	Traffic Engineering
TTL	Time To Live
tunnel	6-in-4 with MPHS
UDP	User Datagram Protocol:
VAD	Voice Activation Detection
VoIP	Voice over IP
VPN	Virtual Private Network
WAN	Wide Area Network
WFQ	Weighted Fair Queuing
WiMAX	Worldwide Interoperability for Microwave Access
without	IPv6 without MPHS
WWW	World Wide Web



**SUATU KERANGKA KERJA TEROPTIMUM BAGI PENINDASAN  
KEPALA TRAFIK IPV6 MASA NYATA DALAM RANGKAIAN  
PENSUISAN LABEL MULTIPROTOKOL (MPLS)**

**ABSTRAK**

Pensuisan Label Multiprotokol (MPLS) dengan IPv6 telah dinyatakan oleh Pasukan Petugas Kejuruteraan Internet (IETF) sebagai mampu diskalakan dan sangat sesuai untuk jenis-jenis trafik yang berlainan seperti VoIP dan Video. Namun, kepala IP yang besar melahirkan overhead kepala yang berlebihan dalam rangkaian MPLS, mengakibatkan kesesakan trafik lalu menjejaskan prestasi rangkaian tulang belakang.

Suatu skema penindasan kepala yang baru, Penindasan Kepala Muatan Multiprotokol (MPHS) dicadangkan dalam tesis ini yang menawarkan penggunaan jalur lebar yang lebih baik bagi MPLS-LSP (Laluan Bersuis Label). Skema yang dicadangkan ini adalah untuk memenuhi ketersediaan yang tinggi bagi rangkaian tulang belakang yang memerlukan skema penindasan yang lebih ringkas. Ianya dibawa merentasi keseluruhan rangkaian tulang belakang, tidak seperti skema yang ada kini yakni, Penindasan Kepala Muatan (PHS) atau Penindasan Kepala Lasak (ROHC), yang kebanyakannya digunakan di rangkaian capaian.

Penyelesaian yang dicadangkan ini membolehkan lebih banyak strim IPv6 masa nyata dan bukan masa nyata pada tulang belakang yang dibolehkan oleh MPLS, disokong dengan QoS (Kualiti Perkhidmatan) hujung-ke-hujung yang boleh diterima. Tesis ini membentangkan empat sumbangan utama terhadap domain

MPLS. Pertama, MPHS menyokong strim IPv6 asli. Kedua, ia membenarkan strim 6-dalam-4 antara Pinggir Pelanggan (CE-keCE). Ketiga, ia menyokong kewujudan bersama trafik bukan masa nyata seperti trafik Sesawang menggunakan aplikasi berasaskan HTTP (Protokol Pindahan Teks Hiper) dan aplikasi berasaskan FTP (Protokol Pindahan Fail). Akhir sekali, MPHS menyokong LSP Eksplisit.

Keberkesanan MPHS telah diuji menggunakan Simulasi Rangkaian versi 2 (NS2). Hasilnya disahkan menerusi model analitis untuk menunjukkan bahawa ia adalah sebanding dan sepadan dengan hasil daripada model simulasi.

Menggunakan MPHS, pertambahan penindasan yang ketara iaitu 64% bagi trafik IPv6 asli dan 63% bagi trafik 6-dalam-4 dilihat sebagai sebanding dengan skema-skema penindasan yang sedia ada. Penggunaan jalur lebar bagi IPv6 masa nyata dan trafik 6-dalam-4 telah meningkat sebanyak 31%. Seterusnya, lengah bingkisan dalam rangkaian MPLS berkurangan daripada sebanyak 22% apabila MPHS diaktifkan di rangkaian teras. Dari segi kesan MPHS terhadap trafik heterogen, masa tindak balas bagi trafik pelanggan-pelayan berkurangan sebanyak 1.7s (daripada 24.88s kepada 23.14s) bagi trafik Sesawang IPv6, menyingkirkan jatuhan bingkisan bagi data UDP. Truput data TCP meningkat sebanyak 20%, meminimumkan masa lengah dengan begitu berkesan. Selain itu, pengurangan kepelbagaian dalam truput bagi trafik TCP dan UDP dapat dilihat apabila MPHS diaktifkan.

# **AN OPTIMIZED FRAMEWORK FOR HEADER SUPPRESSION OF REAL TIME IPv6 TRAFFIC IN MULTIPROTOCOL LABEL SWITCHING (MPLS) NETWORKS**

## **ABSTRACT**

Multiprotocol Label Switching (MPLS) with IPv6 has been defined by the Internet Engineering Task Force (IETF) as highly scalable and well suited for different types of traffic such as VoIP and Video. However, large IP headers create excessive header overhead in a MPLS network leading to traffic congestion degrading the backbone network performance.

A new header suppression scheme, Multiprotocol Payload Header Suppression (MPHS) is proposed in this thesis to offer better bandwidth utilization for MPLS-LSP (Label Switched Path). The proposed scheme caters for high availability of the backbone network that requires much simpler compression schemes. It is carried across the entire backbone network, unlike the existing schemes namely, Payload Header Suppression (PHS) or Robust Header Compression (ROHC), that are mainly used at access network.

The proposed solution allows more real-time and non real-time IPv6 streams over MPLS-enabled backbone to be supported with acceptable end-to-end QoS (Quality of Service). This thesis presents four main contributions over a MPLS domain. Firstly, MPHS supports native IPv6 streams. Secondly, it enables 6-in-4 streams between Customer Edges (CE-to-CE). Thirdly, it supports coexistence of non real-time traffic such as Web traffic using HTTP (Hyper Text Transfer

Protocol)-based applications and FTP (File Transfer Protocol)-based applications. Finally, MPHS supports Explicit LSP.

The effectiveness of MPHS was investigated using Network Simulation version-2 (NS2). The results were validated against analytical models to show that it compares and agrees well with the outcome of the simulation model.

Using MPHS, significant suppression gain of **64%** for native IPv6 traffic and **63%** for 6-in-4 traffic were seen as compared to the existing compression schemes. The bandwidth utilization for real time IPv6 and 6-in-4 traffic was improved by **31%**. Subsequently the packet delay in the MPLS network decreased by **22%** when MPHS was activated at the core network. In terms of effects of MPHS on heterogeneous traffic, response time for client-server traffic improved by **1.7s** (from 24.88 to 23.14) for IPv6 Web traffic eliminating packet drop for UDP data. The TCP data throughput was increased by **20%**, effectively minimizing the delay time. In addition, less variation in throughput for TCP and UDP traffic was seen when MPHS was activated.

# CHAPTER ONE

## INTRODUCTION

### 1.1 Background

The growth of the Internet mirrors the rapid development of new protocols, mechanisms and the remarkable increase of Internet users, for data communication and computer networking. This growth, in turn, has been fueled by the exponential growth of the World Wide Web (WWW). Historically the Web has triggered explosive escalation in the Internet due to fast growth of e-mail after ARPANET (Advanced Research Project Agency Network) establishment (Stallings, 2002). The tremendous increase in traffic volume generated from the Web, real-time multimedia, and multicasting applications has motivated researchers to develop new technologies such as Multiprotocol Label Switching (MPLS), and latest techniques such as header compression that can support better Quality of Service (QoS). However, MPLS is one of the technologies that have been used in Internet backbone.

Data packets for real time applications are mainly a comprised form of voice and video, which represent main drivers for QoS implementation and traffic engineering mechanisms in the internet (Meddeb, 2010). Real time application such as Voice over IP (VoIP) having small payloads as compared to their headers results in significant packet processing overheads (Fortuna & Ricardo, 2009). Header compression schemes such as Robust Header Compression (ROHC) and Payload Header Suppression (PHS) were developed for WiMAX defined in IEEE 802.16, where header overhead of real time traffic is the major concern.

MPLS is a routing and forwarding protocol standardized in 2001. The MPLS architecture is defined in (RFC3031, 2001). The traffic engineered MPLS technology is highly distinguished as the modern approach that guarantees the high level of quality and reliability that we expect from telephony services (Juniper, 2007). Explicit label switched path (Explicit-LSP) is one of the MPLS properties that permit the booking of an explicit LSP that is not necessarily the shortest path. Explicit-LSP can be deployed for different situations, like fast restoration path (in failure cases of node/links), for MPLS-Traffic Engineering usage, load balancing, etc. Internet Service Providers have combined the features of MPLS such as speed, Traffic Engineering (TE), QoS, VPN and resiliency with IPv6 features as an alternative transporting facility over the Internet backbone (Grivaud, 2008).

Typically Internet backbone networks suffer from high load traffic and congestion at edge routers. This bandwidth consumption for packet headers is higher as compared to access network in Wireless, WiMAX or satellite networks. For example, as stated in (RFC4247, 2005), for a real-time application such as VoIP over a WAN, packet headers for 300 million calls per day could consume of about 20-40 Gbps.

Currently there are limited comparative studies between compression and suppression schemes in terms of complexity analysis. Compression schemes are mainly compressing IP header field to less number of bits using encoding and decoding techniques. Suppression schemes work by stripping out an IP header field at the sender node and then restoring the field at the receiver node. No work has been done on implementation of header suppression for IPv6 and 6-in-4 over MPLS for

real time applications. Moreover, Multi-Ingresses and Multi-Egresses in MPLS domains need to be addressed compared to mobile to edge router connection. Further, most compression techniques focuses mainly on wireless and satellite technologies, leaving out other network technologies such as MPLS.

## **1.2 Problem Statement**

One of the most critical aspects in transmitting real time streams over IPv6 networks is the increase in header size in relation to the small payload size compared to IPv4 that represents extra overhead (or successive headers overhead problem). These overheads are considered additional costs in terms of complexity metrics such as time complexity for packet header processing, storage resources such as queuing requirement, and transmission bandwidth requirements. As a result these overheads might contribute negatively in the network performance (QoS) and increase the probability of traffic congestion problem. In terms of QoS parameters, real time applications are very sensitive to the delay and jitter.

In a data stream, most of the fields in the packet header would be the same from the first packet to the last packet. In real time applications such as VoIP, the header overhead problem defined above will be seen in every packet in such a data stream. For example, encapsulated data packet consuming a total of 93 bytes where 60 bytes for RTP/UDP/IPv6 header and 33 bytes for voice data (using GSM 6.0 codec). The header occupies more than half (actually 64%) of the packet size. This overhead problem could be reduced using header suppression scheme that is proposed in this thesis.

### **1.3 Research Motivations**

The following are the motivations for current research:

- The increase in interest over the implementation of MPLS as an efficient transport technology for telecommunication industry.
  
- The need to investigate the effect on performance of large header sizes relative to small data payload for IPv6 Internet Backbone infrastructure using MPLS technologies.
  
- The choice of suitable header reduction scheme to reduce the header overhead for real time data flow.

### **1.4 Objectives**

The overall objective of this thesis is to enhance the efficiency of packet processing in terms of QoS metrics such as bandwidth utilization, throughput, delay, jitter, and packet drop for real time applications. To achieve the above objective, the specific objectives are defined as follows:-

- 1) To perform a qualitative functional comparison between existing header compression schemes (e.g. PHS and ROHC) with the proposed framework for the MPLS-enabled backbone.
  
- 2) To propose a new framework that enhances the QoS performance of real time applications in the MPLS-based backbone by incorporating overhead reduction techniques.



- 3) To analyze the proposed framework using an end-to-end queuing model and study its effect on real time IPv6 traffic by simulation to obtain more statistics regarding the performance.
  
- 4) To investigate the interaction of overhead reduction for real time IPv6 traffic with non real-time traffic such as web traffic using the HTTP protocol, as well as data transfers using FTP protocol.

## 1.5 Scope

The scope of this work (Figure 1.1) is limited to IPv6 client networks for real time traffic, multiple mixed IPv4/IPv6 domains with MPLS support, 6-in-4 tunneling approach, and one real time codec using, **Global System for Mobile Communications (GSM)**.

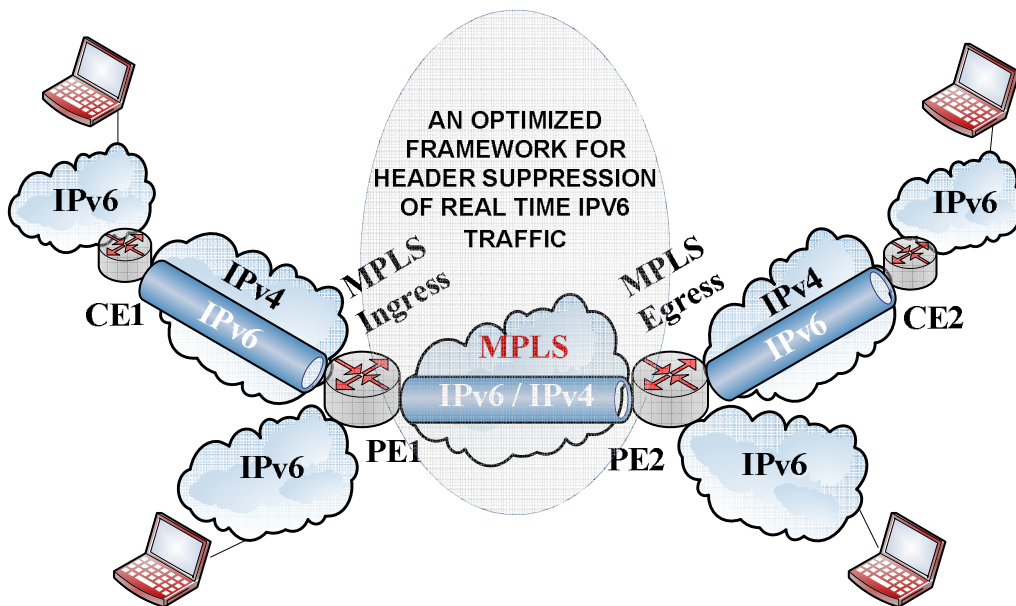


Figure 1.1: Thesis Scope

## 1.6 Research Framework

Fig 1.2 depicts complete research framework of the thesis.

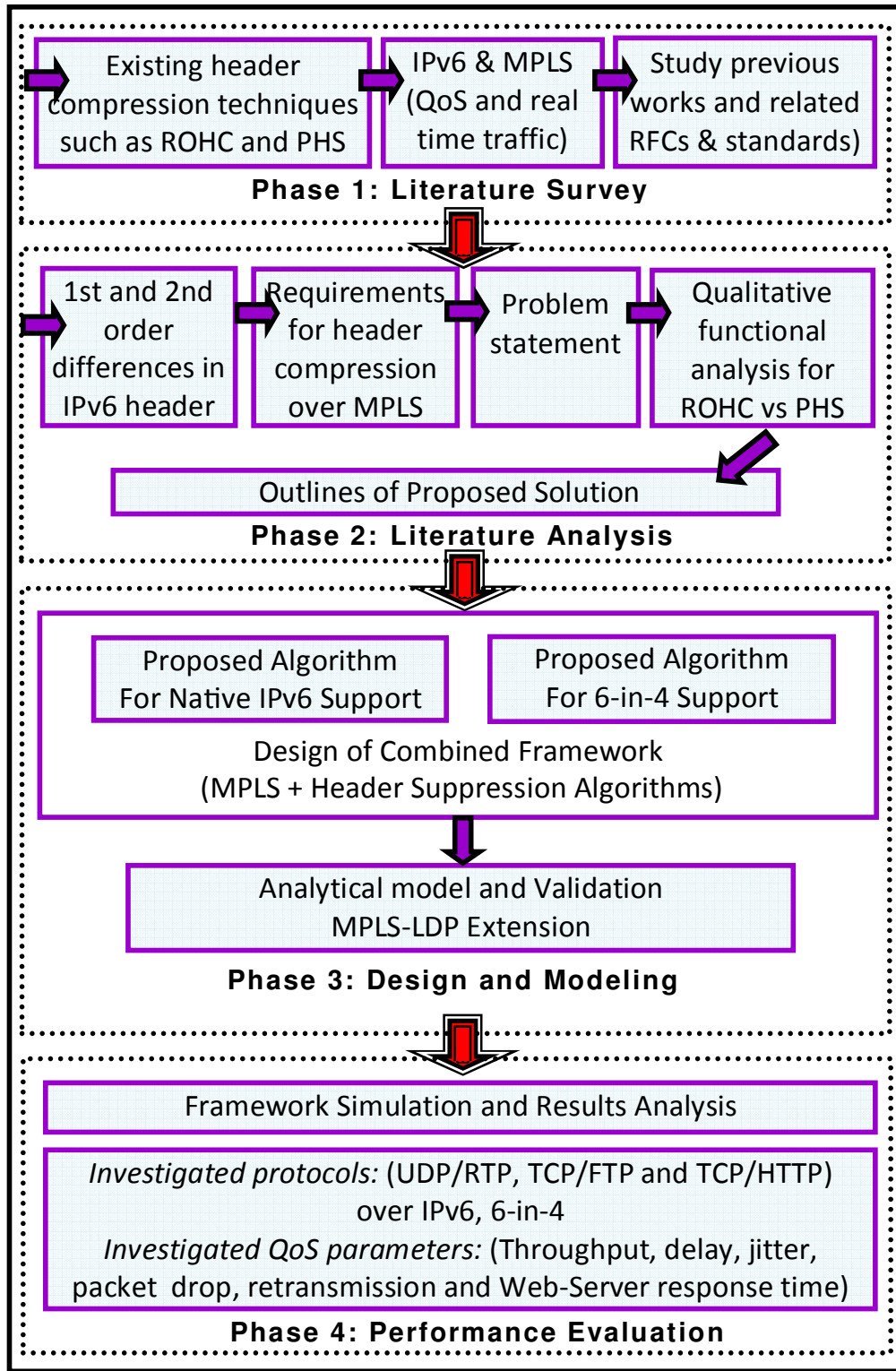


Figure 1.2: Research Framework

## 1.7 Thesis Organization

This thesis is organized into seven chapters.

**Chapter 1** presents the preamble and objectives of this thesis. It starts by presenting a background discussion for the header compression schemes for real time applications and overview for MPLS-IPv6 technology with our research motivations and objectives.

**Chapter 2** extensively covers the literature survey and discusses the most current and related works in header compression research field and MPLS technology. The researcher will also discuss properties of IPv6 header, QoS models, and requirements for header compression over MPLS. Functional analysis for ROHC and PHS is introduced. The reasons of choosing payload header suppression for MPLS framework are discussed.

**Chapter 3** covers the methodology discussion on how the proposed solution was designed. The new header suppression algorithm (MPHS) for MPLS is introduced in this chapter. Functional comparisons for MPHS versus PHS and ROHC using finite state machines (FSM) are introduced. Justifications and consideration for MPHS are also described in this chapter.

**Chapter 4** introduces an end-to-end analytical model for MPHS. It verifies the model by comparing the model and simulation results. It discusses the model in terms of QoS components such as delay, throughput and packet drop.

**Chapter 5** introduces the simulation environment for MPHS in terms of design and analysis. In addition, it states simulation parameters, scenarios for MPHS experiments, and QoS performance metrics used, while simulation results, analysis and discussion for experiments are presented in **Chapter 6**.

**Chapter 7** introduces the research findings; research conclusion, and the possible future work for this study.

## CHAPTER TWO LITERATURE REVIEW

### 2.1 Introduction

This chapter introduces the work related to header compression, real time traffic requirements, MPLS technology, IPv6 and 6-in-4 header specification and principles for QoS. It produces a functional comparison for PHS versus ROHC in terms of complexity. The work includes major and minor research domains originating from QoS IPv6 header compression and supersession (Figure 2.1). In addition, it discusses the most related works in terms of QoS performances, advantages, limitations, and outlines the justification for proposed framework.

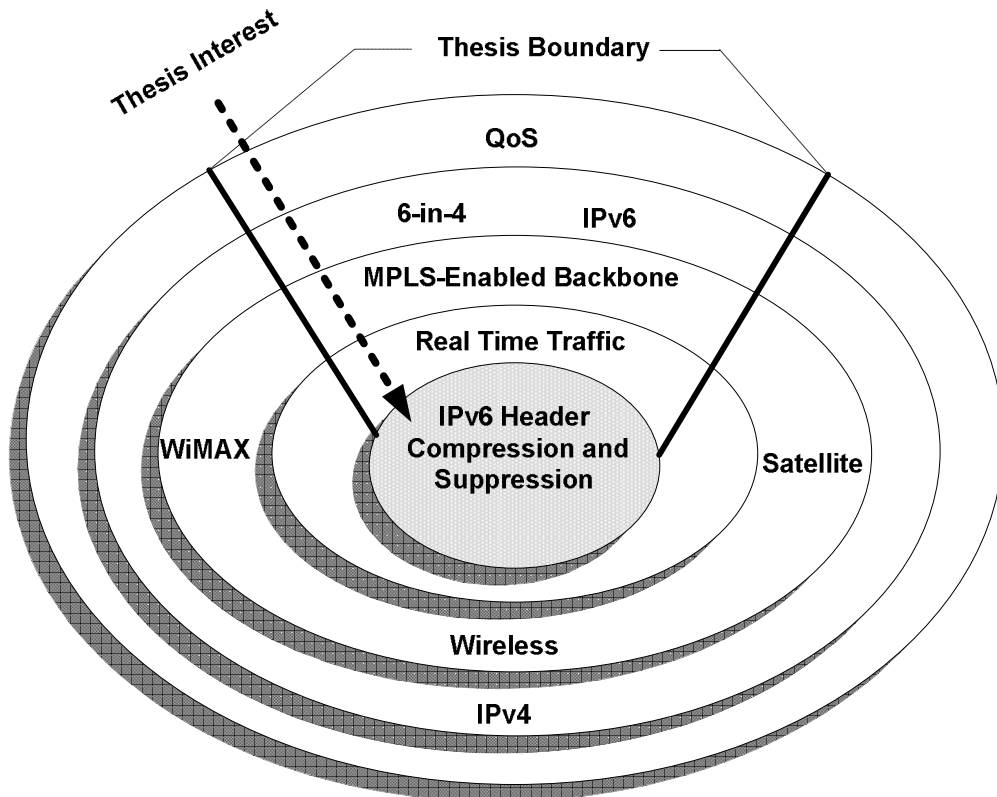


Figure 2.1: Thesis Interest & Boundary

## **2.2 QoS Models, Mechanisms and Metrics**

Internet QoS efforts are aimed to expand the base services of a network to a number of selectable service responses which are distinguished from the best-effort service by supporting superior service level. The expanded services are distinguished by providing a predictable service response, despite varying network traffic load such as the number of concurrent traffic flows (RFC2990, 2000). In terms of quality of service, metrics such as delay, jitter, packet loss, throughput, service availability, and per flow sequence preservation, measures the service quality that IP Traffic experiences.

Real time applications which are mainly voice and video packets representing main drivers for QoS implementation and traffic engineering mechanisms in the internet (Meddeb, 2010). It needs to fulfill certain requirements to achieve end to end QoS metrics. For instance, less than 200ms latency is recommended for voice conversation, at about 30ms jitter is preferred, and below 1 percent packet loss ratio is recommended (Szigeti & Hattingh, 2004).

In terms of OSI (Open System Interconnection) model, Real Time Protocol (RTP) is a transport layer protocol commonly used to transport digitally encoded stream of Voice over IP (VoIP) (RFC3550, 2003) and video over IP. The delivery of the VoIP bearer stream from sender to receiver is a dealing function of RTP which uses one of the signaling protocols such as Session Initiation Protocol (SIP) (RFC3261, 2002) to setup the VoIP session and to determine the codec format used.

Various types of codecs are used to compress and transmit the VoIP packets of real time applications, and most of these codecs produce small packet sizes. Properties of VoIP codec's vary in bandwidth requirements, call delivery quality and complexity. In addition, some codecs use compression to reduce the required bandwidth for a VoIP call, and consequently can be divided into lose and lossless codec's based on compression quality (Evans & Filsfils, 2007). Voice datagram(s) (Taylor, et al., 2005) are on the order of 20 bytes while IPv6/UDP/RTP headers are on the order of 100 bytes. GSM codec is considered as VoIP codec for this thesis.

## **2.2.1 Network QoS Requirements for VoIP Applications**

VoIP applications are defined by different Service Level Agreement (SLA) metric parameters such as (Evans & Filsfils, 2007). The formula for each of the following QoS metric is defined in **Chapter 5**.

### **2.2.1.1 Delay**

Delay is an expression of how much time it takes for a packet of data to get from one designated source to destination. The interactive conversational speech is the main impact factor for one-way end-to-end delay. For example ITU-T recommendation on mouth-to-ear delay was specified in G.114 which uses the E-model and suggested that delay around 150 ms will satisfy for most VoIP applications/users. As the delay increases, the satisfactory level decreases. It will become unacceptable once the delay value passes the 400 ms threshold (delay levels are shown in Table 2.1). The mean opinion score MOS "is a well established scheme, which provides a numeric measure of the quality of a voice call at the destination" (Evans & Filsfils, 2007).

TABLE 2.1: ITU G.114 Determination of the Effects of Absolute Delay by the E-model (Evans & Filsfils, 2007)

<b>Ear-to-mouth</b>	<b>R factor</b>	<b>Objective MOS</b>
<b>Delay &lt; 150 ms</b>	80 – 89	5
<b>150 ms &lt; delay &lt; 250 ms</b>	70 - 79	4
<b>250 ms &lt; delay &lt; 325 ms</b>	60 - 69	3
<b>325 ms &lt; delay &lt; 425 ms</b>	50 – 59	2
<b>Delay &gt; 425</b>	90 - 100	1

Network delay which impacts the VoIP call is one component of the end-to-end delay. In order to satisfy VoIP application requirements, the network QoS design should consider the maximum delay values mentioned above and apportion the budget to various network delay components (such as propagation delay through the backbone such as MPLS backbone, scheduling delay due to congestion, access link serialization delay, and end-system delay due to VoIP codec and jitter buffer).

### **2.2.1.2 Jitter**

It is the parameter that characterizes the variation of network delay. In practice de-jitter buffers (which is used at the destination end-systems to remove jitter or delay variation by converting delay variation to constant delay), and play-out buffers are used to control delay variation.

### **2.2.1.3 Packet Drop**

This QoS parameter is calculated from the difference between the sending number and receiving number of packets. Packet drop could happen as a result of traffic congestion, lower layer errors, network element failures, or loss in the application end-systems.



### 2.2.1.4 Throughput

It is referred to as the amount of data (packet payload) moved successfully from one place (source) to another (destination) in a given time period. Average bandwidth used for a call can be reduced by using various techniques such as Voice Activation Detection (VAD) (which uses silence suppression techniques), and/or header compression techniques such as ROHC or PHS. Goodput is another QoS metric that exclude protocol overhead of transport, network and data link layers from the throughput (Yoo, 2010).

### 2.2.2 QoS Models

In terms of QoS models, (Wallace., 2004) Cisco developed QoS features and categorized them into one of the following three models (Figure 2.2).

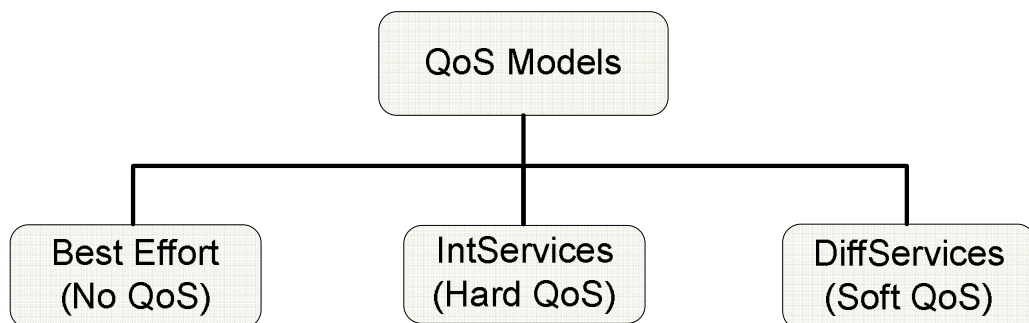


Figure 2.2: QoS Models (Wallace., 2004)

#### 2.2.2.1 Best Effort (No QoS)

Best Effort is the traditional datagram model. No differentiation between elastic and inelastic streams exists in this model which contributes to unpredictable services.

### **2.2.2.2 Integrated Services (Hard QoS)**

IntServ Architecture was defined in (RFC1633, 1994) It is characterized by guaranteeing per-flow QoS and strict bandwidth reservations. IntServ requires signaling for path reservation using Resource Reservation Protocol defined (RFC2205, 1997). Path/RESV messages require admission control and must be configured on every router along the path, and work well on small-scale. The main disadvantages of this model are scaling with large number of flows and requiring devices to retain state information.

### **2.2.2.3 Differentiated Services (Soft QoS)**

DiffServ architecture is defined in (RFC2475, 1998). It is scalable, well supportive to large flows through aggregation, and defines per-hop behavior (PHB). It is capable to create Traffic Conditioning (TC) meaning when edge nodes perform TC such as MPLS ingress nodes, it allows core routers to do more important processing tasks. Additionally, with DiffServ it is tough to predict end-to-end behavior.

DiffServ techniques were designed to integrate in orthogonal manner with traffic engineering mechanisms. While DiffServ techniques are concerned about traffic class's differentiation, traffic engineering needs to ensure QoS provision within class of service.

## **2.2.3 Router Operational Planes**

A router operates in two operational planes: control and data plane. (Evans & Filsfils, 2007):

### **2.2.3.1 Data Plane**

Data plane includes processing intensive functions, packet forwarding lookups and packet filtering functions applied at network nodes and mostly implemented using hardware in high performance routers. QoS mechanisms of data plane can be classified according to the primitive behavior characteristics as follows:

- ❖ Classification (class of service)
- ❖ Marking: Setting up the QoS related traffic fields of IP or MPLS to identify the traffic easily.
- ❖ Policing and shaping: used for maximum rate enforcement.
- ❖ Prioritization: used for setting up traffic priority to provide it with estimated delay and jitter.
- ❖ Minimum rate assurance: Minimum bandwidth assurance for different traffic classes can be achieved by implementing scheduling techniques such as Deficit Round Robin (DRR) and Weighted Fair Queuing (WFQ).

### **2.2.3.2 Control Plane**

Its function includes signaling plane or controlling the data plane. It deals with admission control, routing protocols and resource reservation mechanisms, and it is typically implemented using software. For example RSVP is used for control plane or QoS signaling in the context of integrated services architecture for flow resource reservation & admission control. It is also used to setup MPLS traffic engineering LSPs.

### 2.3 IP Protocols for Internet

IPv4 header contain twelve fields plus option field which yields a size between 20-60 octets compared to eight fields used by IPv6 with a size of 40 octets (Figure 2.3). The option fields in IPv4 headers induce transmission complexities due to the use of variable length headers compared to the fixed header lengths in IPv6. Typically 5-tuple (source address, destination address, protocol, source port, destination port) is used as the IPv4 flow signature. The minimum value of the Internet Header Length (IHL) field of IPv4 is 5 (20 bytes) and the maximum value is 15 (60 bytes), thus at most 40 bytes are found in the option headers. The leftmost 6 bits of the Differentiated services field represent Classes of Service and Priority of Services, for more details refer to (RFC2475, 1998).

Ver = 4 (4 bits)	IP Header Length (4 bits)	Type of Service ( 8 bits)	Total Length (16 bits)	
Identification (16 bits)			Flags (3 bits)	Fragment Offset (13 bits)
Time To Live (8 bits)		Protocol (Value = 41 for 6-in-4) (8 bits)	Header Checksum (16 bits)	
Source IP address (32 bits)				
Destination IP address (32 bits)				
Options (if any) (variable)				

Figure 2.3: IPv4 Header Format

IPv6 header fields are shown in Figure 2.4; these fields occupy 40 octets (bytes) as a fixed length header. Typically 3-tuple of the IPv6 header (IP source address, IP destination address and flow label) representing the IPv6 flow signature.

Traffic class field is equivalent to the Differentiated services field of IPv4 header; it is used for QoS requirements.

Ver = 6 (4 bits)	Traffic class (8 bits)	Flow label (20 bits)		
Payload Length (16 bits)		Next Header (8 bits)	Hop Limit (8 bits)	
Source IP address (128 bits)				
Destination IP address (128 bits)				

Figure 2.4: IPv6 Header Format

Since there is no unique usage for flow label field of IPv6, it is optional and could be deployed for various approaches. In terms of supporting QoS requirements, flow label is used by packet classifiers in order to identify packet's flow. (RFC3697, 2004) declared the specifications and requirements of IPv6 Flow Label values. It specifies that at least 120 seconds (time slot) should exist in order to split the reuse of the same value of flow label for a specific pair of source and destination addresses of IP. It also specifies that Flow label value be set to 0 values by source node for packets that do not belong to any flow.

## 2.4 Multiprotocol Label Switched (MPLS) Technology

MPLS is a routing and forwarding protocol standardized by IETF in 2001. MPLS domain (cloud) is “a contiguous set of nodes which operate MPLS routing and forwarding and which are also in one Routing or Administrative Domain” (RFC3031, 2001).

MPLS facilities in MPLS/VPN, MPLS/QoS, MPLS/TE, ATM + IP and fast rerouting motivated the Internet Service Providers (ISP) to deploy MPLS technology in their IPv4 backbone. These MPLS facilities contributing to minimizing the distortions of streams by setting up multiple LSPs (tunnels) between source and destination to ensure the logical separation between streams (Shekhar Srivastava, Liefvoort, & Medhi, 2009). In terms of failure detection, MPLS supports two levels of failure detection mechanisms, data plane failure detection and control plane failure detection (RFC5884, 2010). This will overcome the condition if only one of the operational planes is working in certain LSP. For example, if the control plane fails but the data plane is working, it will detect it and reroute the traffic to an alternative LSP. The same goes for the case in which the data plane fails.

MPLS networks have the capability of minimizing distortions of streams by setting up multiple label switched paths (LSPs), or tunnels, between source and destination to ensure the logical separation between streams (S. Srivastava, van de Liefvoort, & Medhi, 2009). The success of MPLS technology in providing QoS for real time IP applications makes it one of the favorite choices for ISPs when merged to IPv6 in Internet backbone networks. Therefore several IPv6 scenarios over MPLS have been identified in the literatures as a part of IPv6 deployments (Table 2.2) (Griviaud, 2008).

TABLE 2.2: IPV6 OVER MPLS DEPLOYMENT SCENARIO (GRIVIAUD, 2008)

Scenario	Impact on
<b>IPv6 Tunnels configured on CE</b>	No Impact on MPLS
<b>IPv6 over Circuit_over_MPLS</b>	No Impact on IPv6
<b>IPv6 Provider Edge Router (6PE) over MPLS</b>	No Impact on MPLS core
<b>Native IPv6 MPLS</b>	Require full network upgrade

In terms of VPN over MPLS domain, the similarities in scope and policies simplify the coexistence of IPv6 VPN with IPv4 VPN in MPLS domain. In (RFC4659, 2006), Cisco authored for IPv6 VPN (6VPE) over MPLS/IPv4 infrastructure.

### 2.4.1 MPLS Domain

Ingress and Egress of MPLS domain are two Label Edge Routers (LERs) or Provider Edge routers (PE) representing the input and output doors of the MPLS cloud (Figure 2.5).

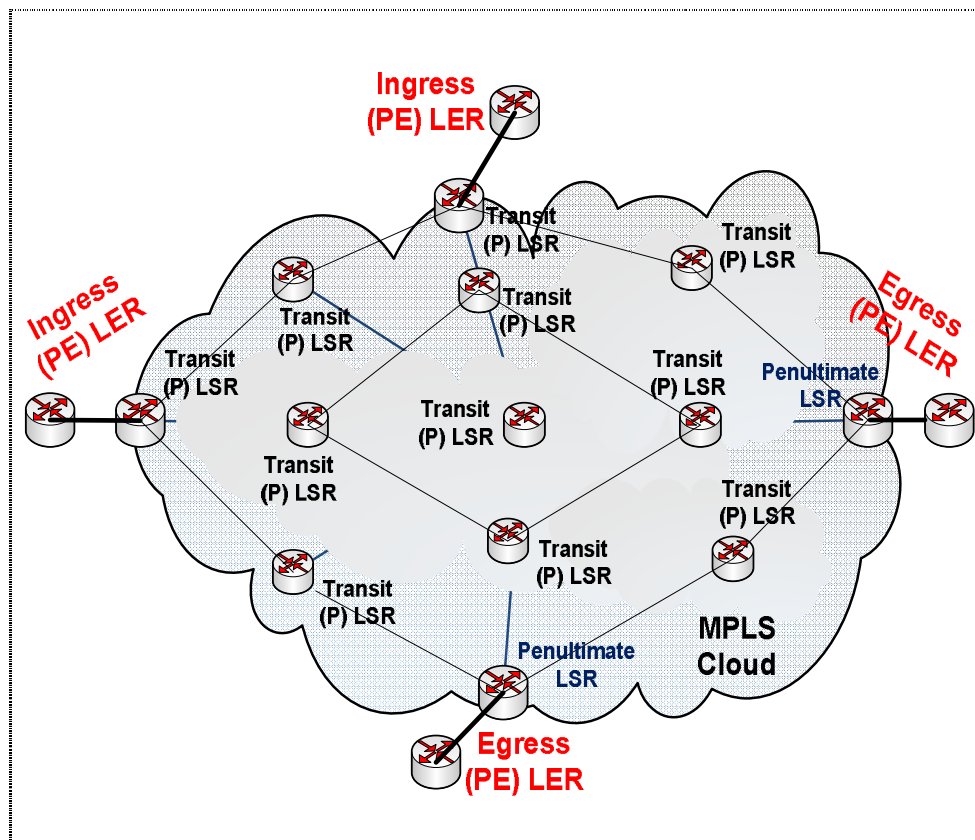


Figure 2.5: MPLS Domain

The MPLS core routers are known as Label Switch Routers (LSRs) or Transit routers or Provider (P) routers. The ingresses and egresses of MPLS cloud are connected via mesh of unidirectional tunnels (paths) namely Label Switched Paths (LSPs).

#### **2.4.2 MPLS Packet Processing and Forwarding Mechanism**

Four main processes are known for dealing with entering packets into MPLS cloud. The first process is the classification process, in which packets entering ingress will be classified and assigned to forwarding equivalent classes (FEC) according to the required treatment similarity, so that the same MPLS label would be provided to all packets belonging to the same FEC. The second is the label push (or encapsulation) process, in which the MPLS label is pushed by ingress to prefix the packet header (Figure 2.6). The third process, forwarding, guides the encapsulated packet through an LSP using a label switching mechanism assisted by the label information base (LIB) table. The fourth is the final label pop (or decapsulation) process, which is maintained by egress (or penultimate) LSR, and followed by a return to normal layer 3 routing (Ghein, 2006; Mine & Lucek, 2008).

The Label Switching Router (LSR) located one hop before the Egress is called **Penultimate**. The Penultimate pops the label instead of Egress, when this facility is activated. Core LSRs forwards the labeled packets without considering their layer 3 IP headers, behaving as Transit routers.