AN EFFICIENT PENDING INTEREST TABLE CONTROL MANAGEMENT IN NAMED DATA NETWORK

RAAID NASUR KADHAM ALUBADY

DOCTOR OF PHILOSOPHY
UNIVERSITI UTARA MALAYSIA
2017

Awang Had Salleh
Graduate School
of Arts And Sciences

Universiti Utara Malaysia

PERAKUAN KERJA TESIS / DISERTASI
(Certification of thesis / dissertation)

Kami, yang bertandatangan, memperakukan bahawa
(We, the undersigned, certify that)

RAAID NASUR KADHAM AL-UBADY

calon untuk Ijazah **PhD**
(candidate for the degree of)

telah mengemukakan tesis / disertasi yang bertajuk:
(has presented his/her thesis / dissertation of the following title):

"AN EFFICIENT PENDING INTEREST TABLE CONTROL MANAGEMENT IN NAMED DATA NETWORK"

seperti yang tercatat di muka surat tajuk dan kulit tesis / disertasi.
(as it appears on the title page and front cover of the thesis / dissertation).

Bahawa tesis/disertasi tersebut boleh diterima dari segi bentuk serta kandungan dan meliputi bidang ilmu dengan memuaskan, sebagaimana yang ditunjukkan oleh calon dalam ujian lisan yang diadakan pada : ***20 Jun 2017.***
That the said thesis/dissertation is acceptable in form and content and displays a satisfactory knowledge of the field of study as demonstrated by the candidate through an oral examination held on: ***June 20, 2017.***

Pengerusi Viva: *(Chairman for VIVA)*	**Assoc. Prof. Dr. Yahya Don**	Tandatangan *(Signature)*
Pemeriksa Luar: *(External Examiner)*	**Prof. Dr. Mahamod Ismail**	Tandatangan *(Signature)*
Pemeriksa Dalam: *(Internal Examiner)*	**Dr. Nur Haryani Zakaria**	Tandatangan *(Signature)*
Nama Penyelia/Penyelia-penyelia: *(Name of Supervisor/Supervisors)*	**Prof. Dr. Suhaidi Hassan**	Tandatangan *(Signature)*
Nama Penyelia/Penyelia-penyelia: *(Name of Supervisor/Supervisors)*	**Dr. Adib Habbal**	Tandatangan *(Signature)*

Tarikh:
(Date) *June 20, 2017*

Permission to Use

In presenting this thesis in fulfilment of the requirements for a postgraduate degree from Universiti Utara Malaysia, I agree that the Universiti Library may make it freely available for inspection. I further agree that permission for the copying of this thesis in any manner, in whole or in part, for scholarly purpose may be granted by my supervisor(s) or, in their absence, by the Dean of Awang Had Salleh Graduate School of Arts and Sciences. It is understood that any copying or publication or use of this thesis or parts thereof for financial gain shall not be allowed without my written permission. It is also understood that due recognition shall be given to me and to Universiti Utara Malaysia for any scholarly use which may be made of any material from my thesis.

Requests for permission to copy or to make other use of materials in this thesis, in whole or in part, should be addressed to:

Dean of Awang Had Salleh Graduate School of Arts and Sciences
UUM College of Arts and Sciences
Universiti Utara Malaysia
06010 UUM Sintok

Abstrak

Perangkaian Data Dinamakan (NDN) adalah seni bina Internet memuncul yang menggunakan model rangkaian komunikasi baharu berdasarkan identiti kandungan Internet. Komponen utamanya iaitu Jadual Minat Tertunda (PIT) menyediakan peranan penting dalam merekodkan maklumat paket Minat yang sedia dihantar tetapi masih menunggu padanan paket Data. Dalam pengurusan PIT, isu pensaizan aliran PIT adalah sangat mencabar kerana penggunaan hayat Minat yang panjang secara meluas terutamanya apabila tiada dasar penggantian yang fleksibel sehingga mempengaruhi prestasi PIT. Matlamat penyelidikan ini adalah untuk mencadangkan satu pendekatan Pengurusan Kawalan PIT (PITCM) yang cekap untuk menangani paket Minat yang mendatang bagi mengurangkan limpahan PIT seterusnya meningkatkan penggunaan dan prestasi PIT. PITCM mengandungi mekanisme PIT Maya Mudah Suai (AVPIT), mekanisme Hayat Minat Ambang Pintar (STIL) dan Polisi Hayat Tertinggi Permintaan Terkecil (HLLR). AVPIT bertanggungjawab mendapatkan ramalan awal limpahan PIT berserta tindakan balasnya. STIL adalah untuk menyesuaikan nilai hayat paket Minat yang mendatang manakala HLLR digunakan untuk menguruskan kemasukan PIT secara cekap. Metodologi penyelidikan khusus diikuti untuk memastikan kerapian kerja bagi mencapai matlamat kajian ini. Perisian simulasi rangkaian digunakan dalam mereka-bentuk dan menilai PITCM. Keputusan kajian menunjukkan bahawa PITCM mengatasi prestasi PIT NDN piawai dengan 45% lebih tinggi kadar kepuasan Minat, 78% lebih rendah kadar penghantaran semula Minat dan 65% penurunan kadar keguguran Minat. Di samping itu, lengahan kepuasan Minat dan panjang PIT dikurangkan dengan ketara masing-masing kepada 33% dan 46%. Sumbangan kajian ini adalah penting dalam pengurusan paket Minat bagi sistem penghalaan dan penghantaran NDN. Mekanisme AVPIT dan STIL serta polisi HLLR boleh digunakan dalam memantau, mengawal dan menguruskan kandungan PIT untuk seni bina Internet masa hadapan.

Kata kunci: Internet Masa Hadapan, Perangkaian Bertumpuan Maklumat, Penghalaan NDN, Pengurusan Giliran Aktif, Simulasi rangkaian.

Abstract

Named Data Networking (NDN) is an emerging Internet architecture that employs a new network communication model based on the identity of Internet content. Its core component, the Pending Interest Table (PIT) serves a significant role of recording Interest packet information which is ready to be sent but in waiting for matching Data packet. In managing PIT, the issue of flow PIT sizing has been very challenging due to massive use of long Interest lifetime particularly when there is no flexible replacement policy, hence affecting PIT performance. The aim of this study is to propose an efficient PIT Control Management (PITCM) approach to be used in handling incoming Interest packets in order to mitigate PIT overflow thus enhancing PIT utilization and performance. PITCM consists of Adaptive Virtual PIT (AVPIT) mechanism, Smart Threshold Interest Lifetime (STIL) mechanism and Highest Lifetime Least Request (HLLR) policy. The AVPIT is responsible for obtaining early PIT overflow prediction and reaction. STIL is meant for adjusting lifetime value for incoming Interest packet while HLLR is utilized for managing PIT entries in efficient manner. A specific research methodology is followed to ensure that the work is rigorous in achieving the aim of the study. The network simulation tool is used to design and evaluate PITCM. The results of study show that PITCM outperforms the performance of standard NDN PIT with 45% higher Interest satisfaction rate, 78% less Interest retransmission rate and 65% less Interest drop rate. In addition, Interest satisfaction delay and PIT length is reduced significantly to 33% and 46%, respectively. The contribution of this study is important for Interest packet management in NDN routing and forwarding systems. The AVPIT and STIL mechanisms as well as the HLLR policy can be used in monitoring, controlling and managing the PIT contents for Internet architecture of the future.

Keywords: Future Internet, Information-Centric Networking, NDN routing, Active Queue Management, Network simulation.

Declaration Associated with This Thesis

Some of the works presented in this thesis have been published or submitted as listed below.

[1] **Raaid Alubady**, Suhaidi Hassan and Adib Habbal, "A Taxonomy of Pending Interest Table Implementation Approaches in Named Data Networking", Journal of Theoretical & Applied Information Technology (JATIT), Vol 91, No.2, pp 411-423, (30 September 2016), ISSN: 1992-8645. [Indexed by SCOPUS]

[2] **Raaid Alubady**, Suhaidi Hassan and Adib Habbal, "Adaptive Interest Packet Lifetime Due to Pending Interest Table Overflow", Advanced Science Letters, Vol. (23), No.(6), pp 5573-5577, (2017), ISSN: 1936-6612. [Indexed by SCOPUS]

[3] Suhaidi Hassan, **Raaid Alubady**, and Adib Habbal, "Performance Evaluation of the Replacement Policies for Pending Interest Table", Journal of Telecommunication Electronic and Computer Engineering , Vol. (8), No. (10), pp 125-131, (2016), ISSN: 2180-1843. [Indexed by SCOPUS]

[4] Suhaidi Hassan, Adib Habbal, **Raaid Alubady**, Mays Salman, "A Taxonomy of Information-Centric Networking Architectures based on Data Routing and Name Resolution Approaches", Journal of Telecommunication Electronic and Computer Engineering , Vol 8, No.10, pp 99-107, (2016), ISSN: 2180-1843. [Indexed by SCOPUS]

[5] **Raaid Alubady**, Suhaidi Hassan and Adib Habbal, "Performance Analysis of Reactive and Proactive Routing Protocols in MANET", Journal of Engineering and Applied Sciences (ARPN), Vol.10, No.3, pp. 1468-1478, (2015), ISSN: 1819-6608. [Indexed by SCOPUS]

[6] **Raaid Alubady**, Suhaidi Hassan and Adib Habbal, "HLLR: Highest Lifetime

Least Request Policy for High Performance Pending Interest Table", The 2016 IEEE Conference on Open Systems (ICOS), pp. 42 - 47, (2016), ISSN: 2473-3660. [Indexed by SCOPUS]

[7] **Raaid Alubady**, Mohammed Al-Samman, Suhaidi Hassan, Suki Arif, Adib Habbal, "Internet Protocol MANET vs Named Data MANET: A Critical Evaluation", Proceedings of the 4th International Conference on Internet Applications, Protocols and Services (NETAPPS2015), Vol. (4), pp.70-76, Putrajaya, Malaysia, (2015).

[8] **Raaid Alubady**, Mays Salman, Suhaidi Hassan and Adib Habbal, "Review of Name Resolution and Data Routing for Information Centric-Networking", Proceedings of the 4th International Conference on Internet Applications, Protocols and Services (NETAPPS2015), Vol. (4), pp.48-53, Putrajaya, Malaysia, (2015).

[9] **Raaid Alubady**, Suhaidi Hassan and Adib Habbal, "Adaptive Virtual Pending Interest Table: Conceptual Model", Proceedings of National Workshop on Future Internet Research (FIRES2016), pp.1-6, (May 2016). Gurun, Kedah, Malaysia.

[10] **Raaid Alubady**, Suhaidi Hassan and Adib Habbal, "The Role of Management Technique for High Performance Pending Interest Table A Survey", Technical Report UUM/CAS/InterNetWorks/TR2017-12, InterNetWorks Research Laboratory, School of Computing, Universiti Utara Malaysia, pp. 1-38, (2017), available online: http://internetworks.my/ [Technical Report].

Acknowledgements

In the name of ALLAH, Most Gracious, Most Merciful:

“Glory be to Thee! We have no knowledge but that which Thou hast taught us; surely Thou art the Knowing, the Wise”. (The Holy Qur’an - (Surah Al Baqarah 2:32))

Peace is upon to Muhammad S.A.W., the messenger sending to guide people in the true way and all praises and thanks goes to almighty ALLAH for giving me the patience, the health and the guidance in completing this thesis successfully.

All praise and glory be to ALLAH for granting me health, strength and knowledge to attain this stage of my life journey. Favors and mercy of preserving me are undeniable. I want to thank so many wonderful and talented people for making my time in Malaysia here at the Universiti Utara Malaysia possible and the chance to do my research at InterNetWorks Research Laboratory (IRL) in School of Computing-Universiti Utara Malaysia.

I will start to gratefully and sincerely thank my supervisor Prof. Dr. Suhaidi Hassan for being my academic supervisor and allowing me to be a member of his research team. Prof. Suhaidi allowed me to go my own ways for my research and his technical knowledge made working with him a privilege. Therefore, I would like him to know that I appreciate all of his help, effort, encourage, support and it was a honor and pleasure to work with him. I say a very big thank you and may ALLAH increase you in wisdom, strength, health and wealth. I would also like to thank my co-supervisor Dr. Adib Habbal for his invaluable contributions through this wonderful journey. He is always found time to help me regardless of his own workload. I consider myself very lucky to have worked with him in this area has broadened my understanding and improved my thinking.

Special appreciation and thanks go to Mr. Alexander Afanasyev, who is a leader in a ndnSIM simulator as well as in NDN environment for his advice and help with all the problems I encountered during my work. Athanasius was always patient with any questions I email him. Furthermore, I would like to extend my appreciation to Mr. Michele Mangili, Mr. Safdar Hussain and Mr. Carlos Anastasiades who have helped me in getting their works in order to understand the area, which is related with my works. In addition, I would be grateful to many other members of the ndnSIM group for insightful discussions that built up my understanding of the Named Data Network architecture design in general and ndnSIM simulator in particular. Also, my deepest gratitude goes to network research community such as ResearchGate, Academia, NS3-users group and STACK overflow group for questions, answers and discussions.

I would like to take this opportunity to thank all my brothers and sisters in InterNet-Works Research Laboratory (IRL) who kept me sane throughout my PhD and who ensured that I play just as hard, if not harder, then I worked. Special thanks particularly Rafid, Atheer, Haider, Salah, Gamal, Omar, Mwafaq, Yousef, ,Ikram, Ibrahim, Walid, Swetha, Sushank, Shivaleela, Wael, Alaa, Mays and all other maybe I forget them.

Finally, my heartiest gratitude goes to my family, to my father my mother, my brothers and my sisters who always have faith in me and prays for my success, to my beloved wife Rana for her understanding, support, and love during this journey, to my children Ali Aldur and Retaj for being so sweet and loving. A special appreciation goes to Dr. Ali Alqaisi, who is helping, support and encourage me before I start my PhD study, and last but not least of all my friends here as well as in my country.

Dedication

For my family . . .

my father;

my mother ; and

my brothers and sisters

my wife Rana; and

our children Ali and Retij

my teachers at all capacity of my knowledge pursuit

Table of Contents

Permission to Use . . . i
Abstrak . . . ii
Abstract . . . iii
Acknowledgements . . . vi
Table of Contents . . . ix
List of Tables . . . xiii
List of Figures . . . xiv
List of Abbreviations . . . xvii

CHAPTER ONE: INTRODUCTION . . . 1
1.1 Information-Centric Networking . . . 2
1.2 Research Motivation . . . 7
1.3 Research Problem . . . 9
1.4 Research Questions . . . 11
1.5 Research Objectives . . . 12
1.6 Scope of the Research . . . 13
1.7 Significance of the Research . . . 14
1.8 Research Steps . . . 15
1.9 Organization of the Thesis . . . 16

CHAPTER TWO: LITERATURE REVIEW . . . 18
2.1 Current Internet Architecture . . . 19
2.2 Future Internet: Information-Centric Networking . . . 21
2.3 Named Data Networking . . . 25
2.4 Classification of Named Data Networking . . . 27
2.4.1 NDN Architecture . . . 28
2.4.1.1 Packets Types in NDN . . . 28
2.4.1.2 Tables Types in NDN . . . 29
2.4.1.3 NDN Lookup and Forwarding Routing Scenario . . . 31
2.4.2 NDN Services . . . 32

2.4.3 NDN Applications 37
2.5 Pending Interest Table 40
2.5.1 PIT Significant Features 41
2.5.2 PIT and Research Issues 43
2.6 PIT Management Techniques 45
2.6.1 PIT Implementation Techniques 46
2.6.1.1 Name Prefix Hash Table (NPHT) 46
2.6.1.2 Fingerprint-only PIT 47
2.6.1.3 Encoded Name Prefix Trie (ENPT) 47
2.6.1.4 Distributed Bloom Filter based PIT (DiPIT) 48
2.6.1.5 Compressed PIT 48
2.6.1.6 MaPIT 49
2.6.2 PIT Placement Techniques 52
2.6.2.1 Input Line Card Placement 53
2.6.2.2 Output Line-Card Placement 53
2.6.2.3 Input/output Line-Card Placement 54
2.6.2.4 Third Part Line Card Placement 54
2.6.3 PIT Replacement Policies 57
2.6.3.1 Persistent Replacement Policy 57
2.6.3.2 Random Replacement Policy 57
2.6.3.3 Least Recently Used Replacement Policy 58
2.6.4 Adaptive Interest Lifetime Techniques 61
2.6.4.1 Dynamic PIT Entry Lifetime (DPEL) 62
2.6.4.2 CCNTimer 63
2.6.4.3 Interest Control Protocol (ICP) 64
2.7 Theories Pertinent to PIT Management 67
2.7.1 Queuing Theory 68
2.7.2 Renewal Theory 69
2.7.3 Scheduling Theory 70
2.8 Overview of Applied Evaluation Methodology 71
2.8.1 ICN Simulators Tools 71
2.8.2 Topology Selection 78

2.9 Summary 82

CHAPTER THREE: RESEARCH METHODOLOGY 83

3.1 Design Research Methodology Framework 83

3.2 Initial Plan of Research 86

3.3 Overview and Critical Analysis 88

3.4 System Modeling 89

3.4.1 Pending Interest Table Control Management Conceptual Model . 90

3.4.2 Adaptive Virtual Pending Interest Table Mechanism 91

3.4.3 Smart Threshold Interest Lifetime Mechanism 94

3.4.4 Highest Lifetime Least Request Policy 96

3.5 Design of Experiments 99

3.5.1 Editor Language 99

3.5.2 Simulation Settings 100

3.6 Verification and Validation 102

3.7 Performance Evaluation 105

3.7.1 Evaluation Metrics 106

3.7.2 Results Analysis and Discussion 108

3.8 Research Conclusion 108

3.9 Research Reporting 109

3.10 Summary 109

CHAPTER FOUR: PENDING INTEREST TABLE CONTROL MANAGEMENT APPROACH 110

4.1 Pending Interest Table Control Management Approach 110

4.2 Adaptive Virtual Pending Interest Table Mechanism 112

4.2.1 Description of AVPIT Mechanism 113

4.2.2 Analysis of AVPIT Mechanism 118

4.2.3 AVPIT Verification and Validation 125

4.2.4 AVPIT Evaluation 128

4.3 Smart Threshold Interest Lifetime Mechanism 130

4.3.1 Description of STIL Mechanism 131

4.3.2 Analysis of STIL Mechanism 132

4.3.3 STIL Verification and Validation . . . 137
4.3.4 STIL Evaluation . . . 144
4.4 Highest Lifetime Least Request Policy . . . 147
4.4.1 Description of HLLR Policy . . . 147
4.4.2 Analysis of HLLR Policy . . . 148
4.4.3 HLLR Verification and Validation . . . 152
4.4.4 HLLR Evaluation . . . 157
4.5 Summary . . . 159

CHAPTER FIVE: SIMULATION ANALYSIS AND EVALUATION . . . 161
5.1 Pending Interest Table Control Management . . . 161
5.2 Performance Evaluation of PITCM . . . 165
5.2.1 Abilene and Rocketfuel Topologies with Various Interest Rate . . 166
5.2.1.1 Abilene Scenarios . . . 166
5.2.1.2 Rocketfuel Scenario . . . 171
5.2.2 Abilene and Rocketfuel Topologies with Various PIT Size 176
5.2.2.1 Abilene Scenario . . . 177
5.2.2.2 Rocketfuel-mapped Scenario . . . 181
5.3 Discussion on PITCM Performance . . . 185
5.4 Summary . . . 187

CHAPTER SIX: CONCLUSION AND FUTURE WORK . . . 188
6.1 Summary of the Thesis . . . 189
6.2 Contributions of the Thesis . . . 192
6.3 Research Limitation . . . 196
6.4 Recommendation for Future Work . . . 196

REFERENCES . . . 199

List of Tables

Table 2.1 PIT Implementation Techniques Summaries 50
Table 2.2 PIT Placement Techniques Summaries 56
Table 2.3 PIT Replacement Techniques Summaries 59
Table 2.4 PIT Entry Lifetime Techniques Summaries 66
Table 2.5 Comparison Between Different Simulations 78
Table 2.6 Validation and Evaluation the Design Models vs Topologies 82
Table 3.1 Summary of the Notation Used in this Thesis 101
Table 3.2 Design Range for the Simulation Parameters 102
Table 4.1 PIT Process After Overflow based on STIL Mechanism 139
Table 4.2 PIT Process After Overflow based on NDN PIT 140
Table 4.3 PIT Process After Overflow based on STIL Mechanism (detected face) 141
Table 4.4 Interest Packet Data Set 153

List of Figures

Figure 1.1	Internet and ICN hourglass Model	4
Figure 1.2	Content Advertisement and Retrieval in NDN	5
Figure 1.3	NDN Router Node and Lookup Process	6
Figure 1.4	Impact of PIT Overflow on Named Data Networking	11
Figure 1.5	Research Scope	13
Figure 2.1	Content Caching in ICN	22
Figure 2.2	ICN Related Architecture Timeline	26
Figure 2.3	Classification of Named Data Networking	27
Figure 2.4	NDN System Architecture	28
Figure 2.5	NDN Packets Types	29
Figure 2.6	NDN Data Structure Tables	30
Figure 2.7	NDN Looking up and Forwarding Scenario	32
Figure 2.8	Classification of NDN Services	33
Figure 2.9	Classification of NDN Application	38
Figure 2.10	PIT Entry Fields	40
Figure 2.11	PIT Management Techniques	45
Figure 2.12	ndnSIM Components Block Diagram	76
Figure 2.13	ICN Simulation tools used	77
Figure 2.14	Dumbbell Topology	79
Figure 2.15	Tree Topology	80
Figure 2.16	Abilene Topology	80
Figure 2.17	Rocketfuel-mapped AT&T Topology	81
Figure 3.1	Research Methodology Framework	85
Figure 3.2	Research Plan	87
Figure 3.3	Main Steps in the Critical Analysis Stage	89
Figure 3.4	PITCM Approach	90
Figure 3.5	Conceptual Model of PITCM	91
Figure 3.6	AVPIT: Reception Interest Block Diagram	92

Figure 3.7 AVPIT: Reception Data Block Diagram 93
Figure 3.8 STIL Block Diagram 95
Figure 3.9 HLLR Block Diagram 97
Figure 3.10 Eclipse C/C++ Development Tools 99
Figure 3.11 CDT's Code Analysis in Eclipse Model 100
Figure 3.12 Verification and Validation Process 103
Figure 3.13 Stages of Verification and Validation 105

Figure 4.1 AVPIT: When VPIT is not Overflow 115
Figure 4.2 AVPIT: When VPIT is Overflow 116
Figure 4.3 PIT Length vs Time for NDN PIT and AVPIT 126
Figure 4.4 Interest Drop vs Time for NDN PIT and AVPIT 127
Figure 4.5 PIT Length vs PIT Size for NDN PIT and AVPIT 129
Figure 4.6 IDR vs No. of Interest Rate for NDN PIT and AVPIT 130
Figure 4.7 Average Entry Lifetime vs PIT Update Process for NDN PIT and STIL with Similar Interest Rate 140
Figure 4.8 Average Entry Lifetime vs PIT Update Process for NDN PIT and STIL with Various Interest Rate 142
Figure 4.9 Average Entry Lifetime vs Time for NDN PIT and STIL 143
Figure 4.10 IRR vs Interest Rate for NDN PIT, DPEL and STIL 145
Figure 4.11 ISR vs Interest Rate for NDN PIT, DPEL and STIL 146
Figure 4.12 PIT Entries 154
Figure 4.13 Entry Replacement based on HLLR, r_c=1 155
Figure 4.14 Entry Replacement based on HLLR, r_c>1 155
Figure 4.15 IRR vs Interests Rate for Persistent and HLLR 156
Figure 4.16 IDR vs PIT Size for Persistent, Random, LRU and HLLR 158
Figure 4.17 ISR vs Interest Rate for Persistent, Random, LRU and HLLR . . . 159

Figure 5.1 PITCM: Test Scenario 164
Figure 5.2 PITCM: Output Results Snapshot 164
Figure 5.3 Abilene Topology: PIT Length vs Interest Rate for NDN PIT and PITCM 167
Figure 5.4 Abilene Topology: IRR vs Interest Rate for NDN PIT and PITCM 168

Figure 5.5 Abilene Topology: IDR vs Interest Rate for NDN PIT and PITCM 169
Figure 5.6 Abilene Topology: ISD vs Interest Rate for NDN PIT and PITCM 170
Figure 5.7 Abilene Topology: ISR vs Interest Rate for NDN PIT and PITCM 171
Figure 5.8 Rocketfuel Topology: PIT length vs Interest Rate for NDN PIT and PITCM 172
Figure 5.9 Rocketfuel Topology: IRR vs Interest Rate for NDN PIT and PITCM 173
Figure 5.10 Rocketfuel Topology: IDR vs Interest Rate for NDN PIT and PITCM 174
Figure 5.11 Rocketfuel Topology: ISD vs Interest Rate for NDN PIT and PITCM 174
Figure 5.12 Rocketfuel Topology: ISR vs Interest Rate for NDN PIT and PITCM 175
Figure 5.13 Abilene Topology: PIT Length vs PIT Size for NDN PIT and PITCM 177
Figure 5.14 Abilene Topology: IRR vs PIT Size for NDN PIT and PITCM . . 178
Figure 5.15 Abilene Topology: IDR vs PIT Size for NDN PIT and PITCM . . 179
Figure 5.16 Abilene Topology: ISR vs PIT Size for NDN PIT and PITCM . . . 180
Figure 5.17 Abilene Topology: ISD vs PIT Size for NDN PIT and PITCM . . . 180
Figure 5.18 Rocketfuel Topology: PIT length vs PIT Size for NDN PIT and PITCM 181
Figure 5.19 Rocketfuel Topology: IRR vs PIT Size for NDN PIT and PITCM . 182
Figure 5.20 Rocketfuel Topology: IDR vs PIT Size for NDN PIT and PITCM . 183
Figure 5.21 Rocketfuel Topology: ISD vs PIT Size for NDN PIT and PITCM . 184
Figure 5.22 Rocketfuel Topology: ISD vs PIT Size for NDN PIT and PITCM . 184

List of Abbreviations

AIMD	Additive Increase Multiplicative Decrease
AQM	Active Queue Management
AVQ	Adaptive Virtual Queue
API	Application Program Interface
AsiaFI	Asia Future Internet BF
AVPIT	Adaptive Virtual Pending Interest Table
AVQ	Adaptive Virtual Queue
BF	Bloom Filter
BGP	Border Gateway Protocol
CBCB	Combined Broadcast and Content-Based
CBF	Counting Bloom Filter
CBN	Content-Based Network
CBR	Constant Bit Rate
CCN	Content Centric Networking
CCNPL-Sim	Content Centric Networking Packet Level Simulator
ccnSim	Content Centric Networking Simulator
CCNx-DCE	Content Centric Networking_x-Direct Code Execution
CDN	Content Delivery Network
CDT	C/C++ Development Tool
CodAn	Code Analysis
COMET	COntent Mediator architecture for content-aware nETworks
CONET	Content-Centric Inter-Network
CRC-32	Cyclic Redundancy Check 32
CS	Content Store
CSMA	Carrier Sense Multiple Access
DDoS	Distributed Denial of Service
DHT	Distributed Hash Table
DiPIT	Distributed Bloom Filter based Pending Interest Table
DNS	Domain Name Service

DONA	Data-Oriented Network Architecture
DPEL	Dynamic Pending Interest Table Entry Lifetime
DRM	Design Research Methodology
E2E	End to End
ECN	Explicit Congestion Notification
ENPT	Encoded Name Prefix Trie
EU FIA	European Future Internet Assembly
FCFS	First Come First Serve
FIB	Forwarding Information Base
FIE	Forwarding Information Entries
FTP	File Transfer Protocol
GCC	GNU C Compiler
GENI	Global Environment for Network Innovations
GPLv2	General Public License, version 2
GUI	Graphical User Interface
HEP	High Energy and unclear Physics
HLLR	Highest Lifetime Least Request
ICN	Information-Centric Networking
IDE	Integrate Development Environment
ICP	Interest Control Protocol
IDR	Interest Drop Rate
INET	Internet NETtworking
IoT	Internet of Things
ISD	Interest Satisfaction Delay
IP	Internet Protocol
IPv4	Internet Protocol version 4
IPv6	Internet Protocol version 6
IRR	Interest Retransmission Rate
ISR	Interest Satisfaction Rate
IT	Index Table
LPM	Longest Prefix Matching

LRU	Least Recently Used
LSPSIN	Line Speed Publish/Subscribe Inter-Networking
LTI	Long-Term Interest
MA	Mapping Array
MBF	Mapping Bloom Filter
NACK	Negative ACKnowledgment
NAT	Network Address Translation
NCE	Name Component Encoding
NDN	Named Data Networking
ndnSIM	Named Data Networking Simulator
NetInf	Network of Information
NOD	Named DataObject
NPHT	Name Prefix Hash Table
NPT	Name Prefix Trie
NS 2	Network Simulation 2
NS 3	Network Simulation 3
NSF	National Science Foundation
OMNeT++	Objective Modular Network Testbed in C++
OPNET	Optimized Network Engineering Tool
P2P	Point to Point
PARC	Palo Alto Research Center
PEs	Propagation Entries
PEL	Pending Interest Table Entry Lifetime
PHT	Propagating Hash Table
PIT	Pending Interest Table
PITCM	Pending Interest Table Control Management
PQ	Priority Queuing
PS	Packet Store
PSIRP	Publish-Subscribe Internet Routing Paradigm
PURSUIT	Publish-Subscribe Internet Technologies
QoS	Quality of Service

RED	Random Early Detection
REM	Random Exponential Marking
RLDRAM	Reduced Latency Dynamic Random Access Memory
RTO	Retransmission TimeOut
RTT	Round-TripTime
SAIL	Scalable and Adaptive Internet Solutions
SAVQ	Stabilized Adaptive Virtual Queue
SRED	Stabilized Random Early Drop
SRAM	Static Random Access Memory
STIL	Smart Threshold Interest Lifetime
SVB	Stabilized Virtual Buffer
TCP	Transmission Control Protocol
TCP/IP	Transmission Control Protocol/Internet Protocol
UBF	United Bloom Filter
UGC	User-Generated Content
UPD	User Datagram Protocol
URL	Uniform Resource Locator
V&V	Verification and Validation
VoD	Video on Demand
VPIT	Virtual Pending Interest Table
WSN	Wireless Sensor Networking

CHAPTER ONE
INTRODUCTION

At the beginning, when the Internet was developed, users were academic in nature; they were merely interested in file transfers, for example, mail exchange [1]. After that, with the develop of new technology, especially the advent of computing devices that have the ability to connect to the Internet, people have more access to the Internet than ever. The old dream of having information at one's fingertips, any place, and any time' is no longer a dream at all. Thus, the generated data traffic has increased at an inconceivable speed and is exhausting network resources, such as available bandwidth and Internet Protocol (IP) address [2]. The popularity of the Internet has caused the data traffic on the Internet to grow dramatically every year during the last several years [3]. The main cause of the Internet growth is to share and distribute information, e.g., academic, social, commercial, mobile video, and cloud computing over the Internet [4].

In the other words, nowadays, users of networks have evolved significantly to be dominated by content distribution and retrieval, while still the basic infrastructure is dependent on the connection between the End-to-End (E2E) of their IP addresses. Access to content and services requires naming methods since methods include the Uniform Resource Locator (URL), which links content to the Internet hosts [5]. On the other hand, the emergence of new applications, such as social networking [6, 7, 8], Video on Demand (VoD) [9, 10], sensor networking [11], Interactive on-line gaming [12, 13] and Internet of Things (IoT) [14, 15, 16], have led the Internet communication to become named content objects rather than on host-location [17].

In brief, the question is to figure out whether the current architecture and its properties will turn into the restricting component of the Internet development and a new applications deployment [18]. In this regard, many research projects that focus on future Internet's architecture design were funded in the past (e.g., US NSF GENI [19], EU FIA [20] and AsiaFI [21]). To overcome these aforementioned challenges, the Information-Centric Networking (ICN) [4], a paradigm for the Internet architecture for the future, has been proposed, making named data, rather than of its physical location, the central element of the architecture communication paradigm.

1.1 Information-Centric Networking

Over the years, the number of people using Internet has escalated [22]. Hence, the Internet has become the most important medium for information exchange and the core communication environment for business relations as well as for social interactions. Millions of people all over the world use the Internet for a daily plethora activities, including finding , information access and exchange, enjoying multimedia communications, selling and/or buying goods, and keeping in touch with friends or families [23]. Therefore, it is expected that the number of devices (e.g., mobile devices, computers, sensors) of the Internet will soon grow to be more than 100 billion [18], which may result in a huge amount of data being requested and transferred over the Internet, reaching approximately 1,000 Zettabyte in 2016 based on the Cisco Virtual Network Index (CVNI) [24]. The success of the Internet has created higher hopes and anticipations for new services and applications. Consequently, the current Internet nowadays may not be able to support sufficiently [18].

Today's applications are characterized in terms of what that the subscriber wants, rather than thinking about where it is located. This new proposed architectural design for the future Internet is termed as Information-Centric Networking (ICN). The ICN paradigm consists of redesigning the future Internet architecture, placing named data rather than host locations (i.e., IP addresses) at the core of the network design.

The design concepts of ICN, as discussed in ([25, 1, 26, 27, 28, 29]), is one of the important efforts of several global future Internet research actions. In such systems, the central paradigm is not E2E correspondence between endpoints, as in the present Internet design. Rather, an expanding interest for profoundly adaptable and proficient content distribution has motivated the advancement of architectures that focus on data objects, their properties, and user's interest for the system to accomplish a reliable and efficient distribution of such objects [30]. While mounting the future Internet architecture, many challenges occur that may attract the attention of the researchers, such as routing scalability, mobility, bandwidth, network management, network security, content protection and privacy, trust models, fast forwarding, performance reliability, intelligent distribution of information, robustness and efficiency [31]. Thus, to cope with these challenges, different Internet architectures of the future emerged under the umbrella of ICN, such as DONA [32], 4WARD-NetInf [33], Content Centric Networking (CCN) [34] and Named Data Networking (NDN) [35].

NDN [35] is a new architecture of the future, which is based on networking basics that are driven by hierarchical content names [36]. Despite the fact that the NDN is a totally novel design based on CCN [34]. NDN mirrors our comprehension of the qualities and restrictions of the present Internet architecture, and the one which might be rolled out via incremental positioning over the existing Internet architecture. According to Jacobson *et al.* in [37], the hourglass architecture of the existing Internet focuses on a universal network layer, named IP, that implements the minimum functionality that is necessary for global interconnectivity. The thin waist of the hourglass enabled the Internet's explosive growth through allowing both upper layer and lower layer technologies to develop independently without any unnecessary constraints. NDN architecture retains the same shape of the hourglass; nevertheless, NDN changes the thin waist to concentrate specifically on content rather than on its location. For more as an illustration, NDN architecture changes the semantic of network communication from delivering a packet to a particular destination address to retrieve content which is identified by a given name (see Figure 1.1).

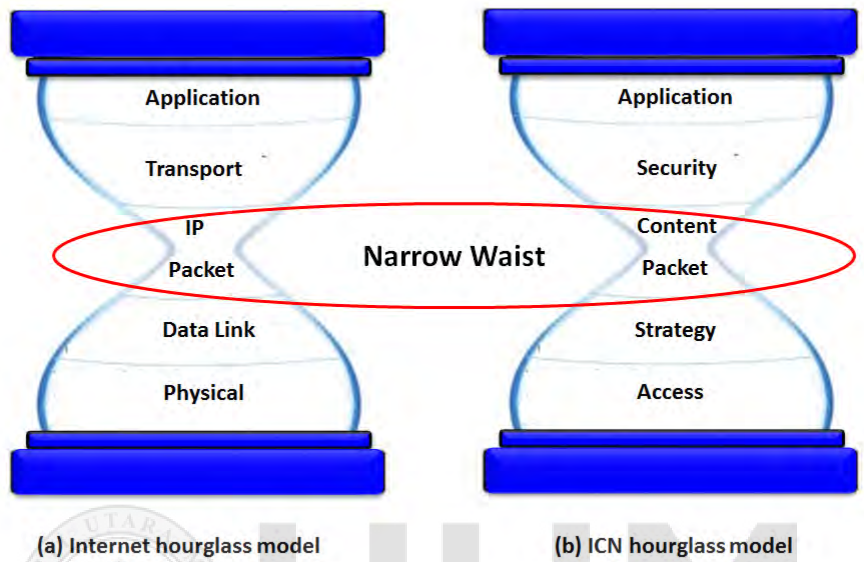


Figure 1.1. Internet and ICN Hourglass Model [38, 39]

NDN is a hierarchical naming mechanism, named content routing and content caching. The hierarchical naming allows the provider's domain name to be used in making routing decisions. In the subscriber-driven NDN, a subscriber sends a request to the network to download a content using its name [37]. The communication in NDN is driven by the users' data, through the exchange of two types of packets: Interest (i.e., request) and Data (i.e., content). Both types of packets carry a name that identifies a piece of data that can be transmitted in one Data packet [40]. An NDN router (i.e., in a traversing path between a publisher and a subscriber see Figure 1.2) can, for instance, cache content objects that traverse it so that subsequent queries for the same objects can be satisfied rapidly by the router. This implies that routing in NDN involves finding and sending copies of data objects to requesting nodes from the most efficient location in the network [26]. The process of content advertisement and retrieval in NDN are illustrated in Figure 1.2.

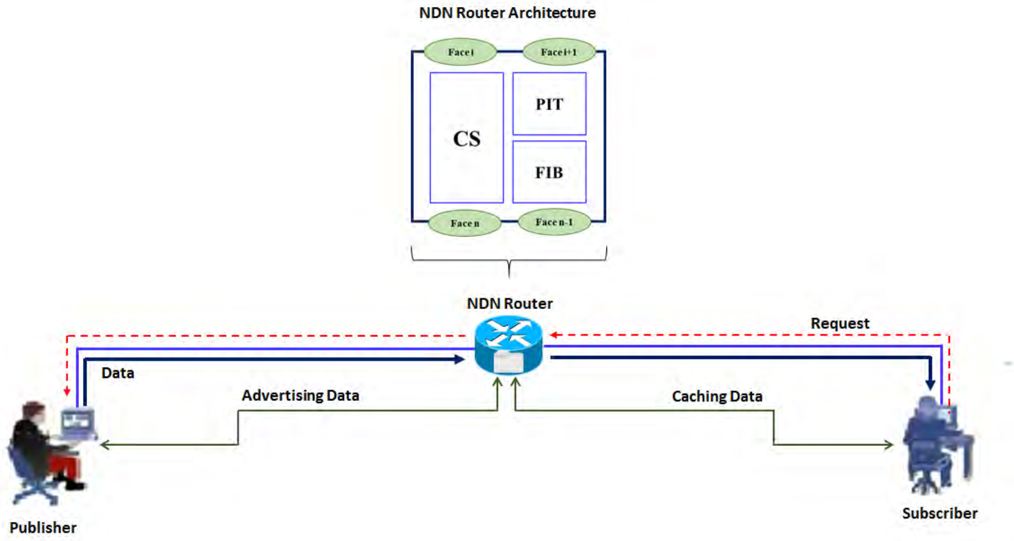


Figure 1.2. Content Advertisement and Retrieval in NDN

In NDN, a publisher does not send content directly to the subscriber, but every data object transmission is regulated by the subscriber. A publisher (also known as the provider of content) that has contents for distribution does not actually send them in the network; rather, the publisher sends advertisement packets to inform the network that it has a content for distribution without any information of the subscriber(s) which might be interested in this content. The subscribers state their interests for a content without having information of the potential publishers that could have accessed the content. Only when the intention of subscribers matches an available content object, the system initiates a transmission path between the publisher and the subscriber; therefore, the content retrieval may start for the intended subscriber(s) [26, 41].

To conduct the above mentioned Interest/Data packet sending capacities, each NDN router maintains three noteworthy data structures tables, including a Forwarding Information Base (FIB), a Pending Interest Table (PIT), and a Content Store (CS) [38, 42, 43, 44, 45, 34]. The FIB is populated by a name-prefix based routing protocol and is used to guide Interests toward data publishers. The PIT caches all Interests which have been sent, but they have not been fulfilled yet. Hence, when an NDN router gets various Interests packets with the same name from downstream subscriber's nodes, it forwards only the first one upstream towards the data publisher. The CS is a temporary cache of Data packets that the NDN routers have received. Since an NDN Data packet is meaningful when it is independent of where it originated from or where it is sent to, it can be stored to fulfill the future Interests. Figure 1.3 demonstrates the NDN router and the query procedure.

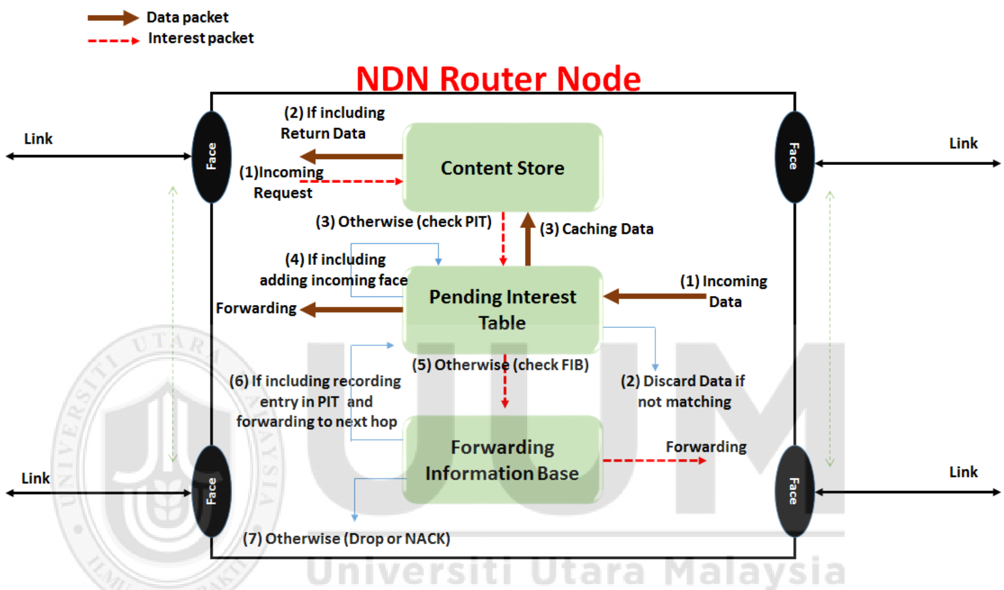


Figure 1.3. NDN Router Node and Lookup Process

PIT is a novel data structure in the context of NDN router design [42]. It is used to keep track of pending as well as the satisfied Interests. PIT brings NDN router significant features (e.g., multipath routing, transmission without the information of destination or source, better security, loop and packet loss detection) [46]. PIT corresponds to the set of active Interest packets forwarded by the subscriber(s) and waiting for the corresponding Data packets to be delivered. Each PIT entry tracks the incoming faces for the Interest packets and is represented at minimum with the tuple "<<name, in-face(s), out-face(s), Lifetime, strategy name>>" [47].

PIT entries are made per content instead of per request; any subsequent Interest packet for an active PIT entry is stifled at the nearby subscriber node, and the present PIT entry is updated with the incoming face information. Since the PIT size is limited, each entry in the PIT is connected with a timeout that depends on the Round Trip Time (RTT) evaluation (i.e., "*RTT describes the lifetime of Interests inside the PIT*" [48]). PIT entry timeouts give a nearby subscriber node the capacity for speedy path recovery by retransmitting the Interest packets over other accessible faces. Whenever a NDN router gets a coordinating Data packet for a pending Interest packet, the obtained Data packet is sent along the face that is indicated by the PIT entry before the request is evicted from the PIT [49].

Although this table introduces many significant features, the size of PIT is a critical challenge [50, 51]. In spite of its feasibility in the current technology, it is challenged by several issues, such as PIT scalability (as discussed in [50, 52, 53, 54, 51]), PIT lookup (as discussed in [55, 56]), PIT filtering (as discussed in [57, 58]), PIT overflow (as discussed in [59, 60, 58]) and Interest lifetime (as discussed in [61, 62, 35]). As a matter of course, these drawbacks motivate the research groups to design an alternate PIT in NDN.

1.2 Research Motivation

NDN is a new architecture for the future Internet that changes the technical protocols in order to support applications with implications for social, economic, and policy dimensions of today's Internet ecosystem [63]. A lot of studies focused on caching, applications and security to make NDN perfect [42, 64]. However, the management of PIT is still one of the primary concerns of "*high-speed forwarding*" that is a core component in NDN [65]. Pending Interest Table is a cache table for Interest packets that requests contents to be transmitted to the subscriber node. In NDN architecture, when NDN router receives an Interest packet, it is stored in PIT [66]. Consequently, the number of PIT entries will increase as the number of Interest packet that arrives, increases. In reality, this table can overflow with consequent service disruption and a possible network failure. Therefore, the management of PIT design is challenging due to its requirements, i.e., per-packet updates, replacement entry condition, look up speed and keep track of the currently unsatisfied Interest packets.

Based on what mentioned above, the influential problem in PIT is two-folded: the size of PIT itself and the amount of entries in the PIT when the number of Interest increases in the network [51]. Thus, the number of Interests that should be stored in the PIT in order to fully utilize the network depends on the link capacity, the size of Data packets and the lifetime of entries inside the PIT [48]. The arrangement of PIT was introduced in the following studies [67, 52, 68, 46, 54] and proposed the implementation of PIT in an efficient way with the purpose of reducing the required memory size. However, these works focus only on the implementation of PIT without any consideration of the PIT overflow problem and management of PIT as a function of content request workload.

The studies in [61, 69, 70, 71] proposed adaptive Interest lifetime techniques in order to adjust the Interest lifetime and regulate the Interest packet rate at the receiver. However, these works do not evaluate the number of unnecessary retransmissions due to too short Interest lifetimes, and as result increases the delay. Finally, the replacement policy may have a significant influence on the cache performance [72] since replacement policies play an important role in memory management [73]. However, these policies did not consider the entry lifetime as a factor to determine the PIT entry to be replaced, since the spent time for each entry of the PIT memory can affect the PIT utilization [74].

The motivation behind this work is that the PIT designing and implementation have not got much interest till now because PIT is a new network component that is not present in any other ICN proposal [68, 75]. In addition, PIT design is challenging because it requires per-packet updates, and the requests stored in the PIT are long, requiring more memory [52]. This represents the main reason for focusing on this component (i.e., PIT) in this research as it plays a very important role in the performance of the NDN routers.

1.3 Research Problem

PIT is the fundamental structure used to maintain the state of each active flow. PIT grows with subscribers sending their Interest packets and shrinks when Data packets are received at the in-network NDN router [26]. Since the access speed required for such a structure and the possibilities offered by the current memory technologies are limited in size, the PIT size might cause a bottleneck [58]. Particularly, when the rate of the created PIT entries is greater than the rate of entries that is consumed [59]. If the network is large with massive requested contents, the PIT size (i.e., the maximum number of pending Interests currently in the PIT) will grow rapidly till the content request is satisfied [26], otherwise it is possible that the PIT will become an overflow. On the other hand, the PIT receives and removes Interest packets exponentially. Hence, it is not easy to predict that the PIT is overflowed via the incoming Interest rate is high [60], especially when the requested Data packet experiences some propagation delay. If the PIT is overflowed, Interest packets will be dropped from the NDN routers. Based on this, subscribers will experience an increasing Interest packet retransmission rate that will cause unnecessary traffic load with consequent slow down or complete collapse in the network [58].

Depending on the PIT status, PIT entry is associated with a timeout that is based on the estimation of an Interest packet lifetime [49]. The Interest packet lifetime parameter is selected by subscribers not under a network controller itself (i.e., based on the available NDN rules). Hence, PIT size may become more complicated by a massive usage of a long Interest packet lifetime that will increase the number of simultaneous entries in the PIT [58]. In other words, when the subscriber is setting long Interest packet lifetime, the entry of Interest packet will stay in the PIT for a long time. Thus, it can increase the probability of discarding incoming Interest packets and leading to increase the Interest packet retransmission. On the other hand, entries with a short lifetime imply a large number of unnecessary Interest packet retransmission, because of delayed Data packets arriving after PIT entries-expired are discarded [69, 70]. Consequently, the PIT size is a crucial factor that can have a huge impact on the number of both satisfied and Interests packet lifetime, as well as the number of packet drops [5].

NDN router maintains a limited space as a cache because of performance and cost. Content replacement policies are used to change the new content with the existing content in the cache [38]. PIT replacement policy is considered as one of the key parameters to measure the effectiveness of the PIT [76]. The NDN routers' prototype implementation (i,e., CCNx) has proposed numerous replacement policies, such as LRU, Random, and Persistent [47]. However, these policies are not efficient via heavy load since they may cause many unnecessary cache replacements. Moreover, these policies do not give any priority to the entry that needs to replace in the PIT when it is filled. As a result, the subscribers would not be able to receive their requested content without delay. In addition, Interests packets requesting non-popular data remain a long time, and would be a large portion in the PIT over the whole network [51]. Therefore, the needs to develop a new content replacement policy has become so crucial with the existence of many content replacement policies, but the efficient policy may never be achieved [38]. The impact of PIT overflow is demonstrated in Figure 1.4.

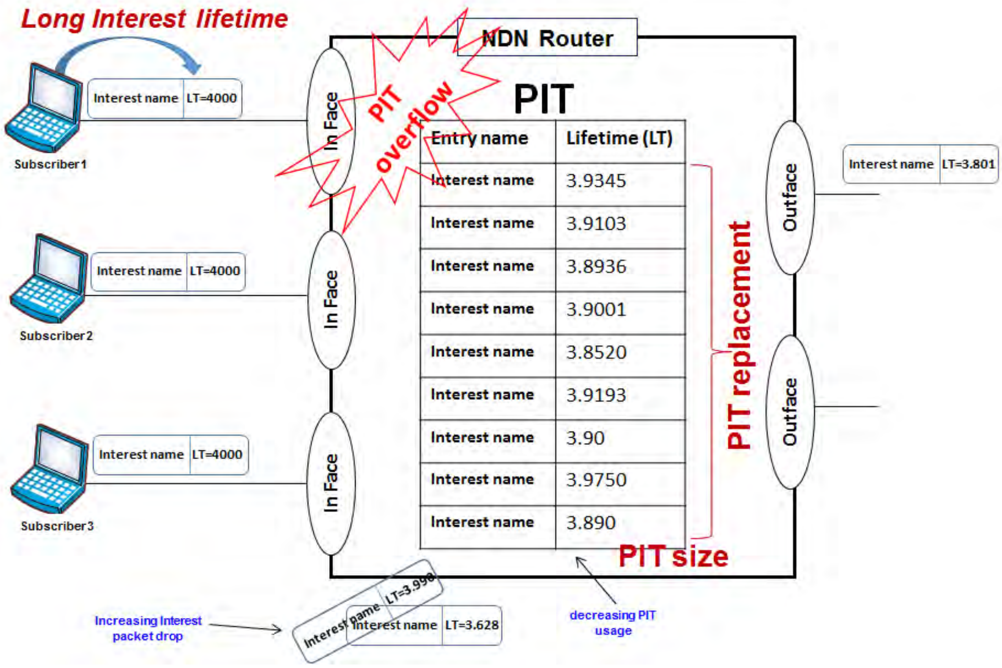


Figure 1.4. Impact of PIT Overflow on Named Data Networking

Based on mentioned above, the influential PIT overflow problem leads to increase the probability of increasing retransmission Interest packets by subscribers, increasing Interest packet drop, increasing Interest satisfaction delay as well as decreasing the utilization of the PIT. These thus, have an effect directly on the performance of PIT. As a result, previous works focus on data structures design or adaptive Interest Lifetime treatment under the threat of PIT state explosion. The proposed approach, Pending Interest Table Control Management (PITCM), develops management of the PIT dynamics as a function of content request workload and network conditions.

1.4 Research Questions

In addressing the overflowing PIT in the NDN architecture, the research tries to answer the following questions.

i. How the overflow in the PIT is predicted when the number of entries grows due to the high packet arrival rate to the PIT?

ii. How the Interest packet lifetime in the PIT adapts to the impending PIT overflows without increasing Interest packet retransmission?
iii. How to determine which PIT entry to be replaced based on PIT conditions for efficient PIT utilization?
iv. How is the proposed approach evaluated to ensure the impact of performance in the NDN environment?

1.5 Research Objectives

The main aim of this research is to develop and deploy a Pending Interest Table Control Management (PITCM) approach in Named Data Network in order to enhance the management operations of PIT. This approach will be monitoring and controlling the incoming Interest packets as well as managing the PIT entries in order to mitigate the PIT overflow. This mitigates the Interest drop, delay and the unnecessary Interest retransmission as well as increases the utilization of PIT and the network overall. This aim can be further achieved by the following particular research objectives:

i. To design an adaptive mechanism that predicts the impending overflow of the Pending Interest Table.
ii. To design a mechanism that adjusts the Interest packet lifetime in order to avoid overstaying of PIT entries during heavy network load.
iii. To design a replacement policy that determines PIT entry to be changed for efficient PIT utilization.
iv. To compile the AVPIT, STIL and HLLR into PITCM in order to evaluate the impact of proposed approach performance in the NDN environment.

1.6 Scope of the Research

Internet communication paradigm for the future is moving from host-centric network to content-centric network. Under this movement, the location where the content is located is no more important, rather the content itself is the most important. NDN is proposed as a solution for architecture to support the content-centric networking by using its Interest packets to request a content and forward Data packets reply to the Interest packet. However, there are still many challenges and solutions to be developed and deployment, PIT management becomes a crucial challenge for the storage of content in NDN routers cache, particularly in PIT.

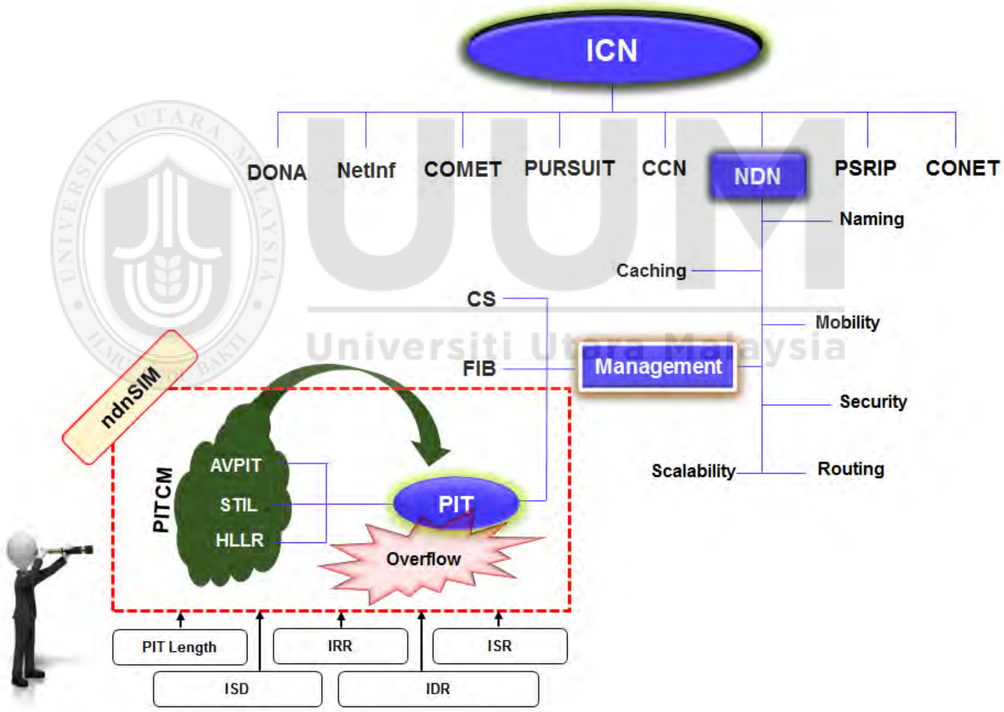


Figure 1.5. Research Scope

The scope of this study is demonstrated in Figure 1.5. PITCM approach consists of three components that can affect the PIT performance based on the NDN architecture, which are AVPIT mechanism, STIL mechanism and HLLR policy. In addition, this research evaluates the proposed PITCM for NDN architecture using an ndnSIM simulator with regard to several performance metrics, such as PIT length, Interest drops rate, Interest retransmission rate, Interest satisfaction delay and Interest satisfaction rate. On the other hand, other NDN issues, such as, PIT implementation, PIT placement, Interest filtering and forwarding strategy are out of this research scope.

1.7 Significance of the Research

The main purpose of this work is to propose a management approach (PITCM) which keeps the initial features of the NDN, and can be implemented by using the current available hardware and software technologies. PITCM is capable of mitigating PIT overflow due to the high packet arrival rate to the PIT. The outputs of this research can provide reliable data transfer features over NDN environment, leading to a set of acceptable performance levels that are required by the subscribers and publishers. In addition, an PITCM has the ability to save the future designs of physical network devices, save time (i.e., delay), bandwidth usage, and as a result could be increased in the utilization of the network overall. Moreover, this study will be also significantly helpful in the emergency applications, for example, natural disaster (i.e., flooding disaster).

According to Shimazu *et al.* in [77],the emergency response delay is one of the major issues that can disable responders (emergency operation centers or emergency rescue teams) to make timely, informed decisions. The main cause of that is because the citizens and emergency teams or emergency operation centers send and receive a huge Interest and Data packet between them. Thus, a lot of the Interest packet will drop because of the PIT overflow (all requests are sent in the same place) and increase escalating crisis. Therefore, implementing PITCM will be very much useful for emergency operation centers, emergency rescue teams, and citizens. Furthermore, it will increase the network utilization, and limit packet loss and delay. The other purpose is to encourage other researchers to focus on the study in ICN generally and NDN in particular, as well as in the PIT because it did not get much attention from the researchers. This is due to that the PIT is a new component of the network, which is not present in any other ICN architecture or current Internet architecture.

1.8 Research Steps

To achieve the objectives of this study, the following research steps were performed:

i. Overview the background literature on PIT management proposals related issues.

ii. Classifying the existing PIT management into four classes based on the direction which is used to manage the PIT.

iii. A comparative performance study of the current PIT management techniques to explore their strengths and weaknesses.

iv. Perform an in-depth study on identify research concept, research problem, research objectives and research scope.

v. Proposing the Adaptive Virtual PIT (AVPIT) mechanism, Smart Threshold Interest Lifetime (SILT) mechanism and Highest Lifetime Least Request (HLLR) policy, which leads to enhance the PIT performance significantly.

vi. Simulate the proposed mechanisms individually in ndnSIM simulator environment and obtain the desired result.

vii. Integrating AVPIT, STIL and HLLR into PITCM approach.

viii. Comparative performance evaluation of the current standard PIT management and the proposed PITCM approach to investigate their contributions and limitations.

1.9 Organization of the Thesis

The thesis is divided into six chapters. Each chapter begins with a short introduction that underlines the key contributions and offers an impression of the chapter. The summaries of the chapters are as follows:

Chapter One presents a general introduction about the research area. It demonstrates the problems related to this study and underlines the significance of the research. Specifically, it presents a background to the importance of PIT in NDN router and the motivation for proposing a new management approach for PIT. Its relevancy is described through importance, motivation, problem statement, research questions, objectives and significance of the proposed research.

Chapter Two specifies a technical background of the research work by reviewing the core properties of the NDN architecture that are essential to describe the proposed work. It presents brief descriptions of the principles and operations of PIT regarding the PIT management techniques. Moreover, it critically evaluates the related works and other studies in the areas associated with the research scope.

Chapter Three explains in general the details about the methodology and research design that is used to achieve the research objectives. In addition, This chapter presents several mechanisms used that combined in order to propose and implement PITCM. The experimental design, verification, validation, and evaluation were also put forward in this chapter.

Chapter Four elaborates the design of the proposed approach. It describes the full deployment and techniques of achieving the PITCM components (AVPIT, STIL and HLLR), including the verification, validation and evaluation of each one. This answers the question of proposing AVPIT, STIL, and HLLR. Finally, the results and analysis are considered briefly in their dedicated sections.

Chapter Five develops PITCM and presents in detail the comprehensive evaluation of the proposed approach through simulations. The chapter examines the PITCM with different simulation scenarios with respect to different network topologies. In addition, the theoretical and graphical comparisons of PITCM with the current solution are also provided in this chapter.

Chapter Six states the conclusion and reviews how the primary research goals have been achieved and highlights the main contributions of the thesis, as well as outlines some possible directions for future work based on the findings of the current study.

CHAPTER TWO
LITERATURE REVIEW

Recent trends in Internet usage and operability of users have diversified into the need for contents instead of location based. This huge dimension necessitated the creation of an approach that is mostly driven towards content-centric to access web, videos on demands, gaming, social interactions and networking and IP-Television. Hence, the future Internet is projected towards slightly changing current semantics to partly dissociate the host of information from the content information. Consequently, several research projects from national and multinational labs, and research agencies proposed a re-branding of the current Internet architecture around the named data or content-centric approach. Therefore, the new proposed architectural design for the future Internet, termed as "Named Data Networking (NDN)", is a joint research effort in developing the new Internet architecture. These research efforts are commonly identified under the auspices of NDN.

The research plan was clarified in Chapter One, whereas Chapter Two describes in greater details the background and several important research related issues of Pending Interest Table (PIT) which is studied, implemented, and managed. These probably assist in defining the general framework of this research. In addition, this chapter focuses on analyzing the NDN communication system, particularly the architecture, features and application. Moreover, a deep analysis and a critical reviewing of the PIT management based on PIT implementation, PIT placement, PIT replacement and adaptive Interest packet lifetime techniques were covered. Based on the above mentioned approaches, this chapter is organized as follows. Section 2.1 is a brief description of the current Internet architecture, whereas a basic concept in Information-Centric Networks is presented in Section 2.2. Section 2.3 and Section 2.4 describe in details NDN that is related to this research scope. PIT is described in Section 2.5. Subsequently, it is followed by Section 2.6 which introduces the main techniques that are used to manage PIT. Theories pertinent to PIT management are described in Section 2.7. Section 2.8 presents the overview of applied evaluation methodology. Finally, the chapter ends with a summary which is presented in Section 2.9.

2.1 Current Internet Architecture

As previously mentioned, the Internet is seen as the packet-switched network on a larger scale due to its coverage. Practically, the packets are usually forwarded using the best-effort service model that is offered by the Internet Protocol (IP) [1]. As a network layer protocol, the IP is responsible of the packet delivery across the entire network. In addition, IP communication is known as a sender-driven concept, where the sender constructs an IP packet, appending both the source and the destination addresses into the packet header before sending it to the network [78]. Currently, the leading usages of the Internet have been shifted to content generation, access sharing and access to media information which could be reported as a trend shift of the initial Internet submission [79].

The location of content, resources, servers, services or users must not be an issue of concern. Therefore, one among several research challenges is how to make communications networks as a location-independent as possible. In addition, due to the high aspiration to make applications more distributed with a better interactivity, coupled with the increasing growth in content demands, several traditional network functions of addressing, naming, routing and forwarding become strongly influenced by the current applications and the content/media in question [80]. Nevertheless, sometimes, it is predicted that the architecture suffers on multiple documented and highlighted problems, many of which could result from the usage engraved to the Internet drastically. For example, the current Internet architecture was designed to provide basic access to specific nodes in a network that is not so dispersed similar to today's usage. The current trend is dominated by information dissemination through the medium of the information requester having to care about the source producing the accessed information as its location. But what happens as [81]:

- Billions of devices are connected; capable of being interrogated and providing the needed information which can be aggregated into services?
- Users currently demand image resolutions in video which requires bandwidths greater than what it can support over typical link lengths using the current physical link layers?
- More connected users' process delay in critical real time video and audio communications while using the Internet?

These changes will only be supported in the existing Internet through huge investment, and even then the architecture may be exhibited some unstable characteristics. According to Mathieu *et al.* in [27], the relationships between network entities are not limited to the view of the network topology, but the network entities epitomize social or content-aware connections between users, which can share common interests (e.g., online photo, newsgroups, and video sharing service including social networks such as YouTube, Twitter or Facebook). The results of graph model of today's Internet communications is very complex. Nodes are generally pieces of data instead of endpoints addressed by the underlying IP protocol; therefore, they are linked by one or more kinds of interdependency based on the idea of intent, interest or policy-based membership. While still overlaying on top of the host-to-host conversation model, all the current workarounds for Internet support of emerging content-centric applications increase the complexity and do not efficiently map all the relevant ties between the nodes in terms of multicast delivering, scalability and Quality of Services (QoS) guarantees, mobility and security.

To cope with the aforementioned challenges, the incompatibility between the communication and usage models of the Internet, Information-Centric Networking (ICN) [1], that is referred to as content-centric or a data-driven networking paradigm proposed earlier. ICN serves as a "clean slate" readjusted architecture through its redesigning principles of placing content in the foreground, with wider interest to the content ahead of its location in all network transactions. Following these submissions, the thesis refers to the terms information and data as it relates to content interchangeably.

2.2 Future Internet: Information-Centric Networking

Of course, the Internet architecture for the future will not only bring new content and media. It should be aimed to retain some of the current semantics, but with added functionalities of how data would be handled. Also, attention requires to be paid to the privacy policies, presentation, services, communications, and infrastructure, which are essential building constitute of the Internet architecture for the future [81]. Therefore, the new generation of the Internet namely Information-Centric Networking (ICN) [82, 83, 84, 1, 43, 34] introduced a radical change in Internet communications. ICN emphasizes information access regardless of location through a new data-based approach, allowing networks to actively deliver content. Since an information plays the key role in an ICN, the information model is required to represent information appropriately and support efficient information dissemination [33]. The main principle is that a communications network must allow subscribers to focus on content they require instead of having a specific reference, physical location from which the content is to be retrieved. ICN employs innovative concepts, such as named content, name-based routing, and security mechanisms that are applied directly to content and in-network content caching [1].

In other words, ICN is a longer-term approach to respond to the current Internet requirements, aiming at incorporating right at its core functionality to support the routing-by-name paradigm [80]. Hence, the ICN approach is built with the communication style that concentrates more around the provision and consumption of communicated information to match Internet users' interest. The major concern of this network paradigm is to explore and find other forms of delivering information with more interest paid to content more than the end-point of the information [26]. From ICN perspectives, the IP addressing in the traditional Internet is going to be fully replaced with the new naming scheme for better identification of content objects. Consequently, this will partly dissociate the host locations and the main content identifiers as seen in Uniform Resource Locator (URL) as an example. Naming content object in particular will directly provide a more globally unique placement and grant the total independent of the content location (i.e., the host holding the data) [27].

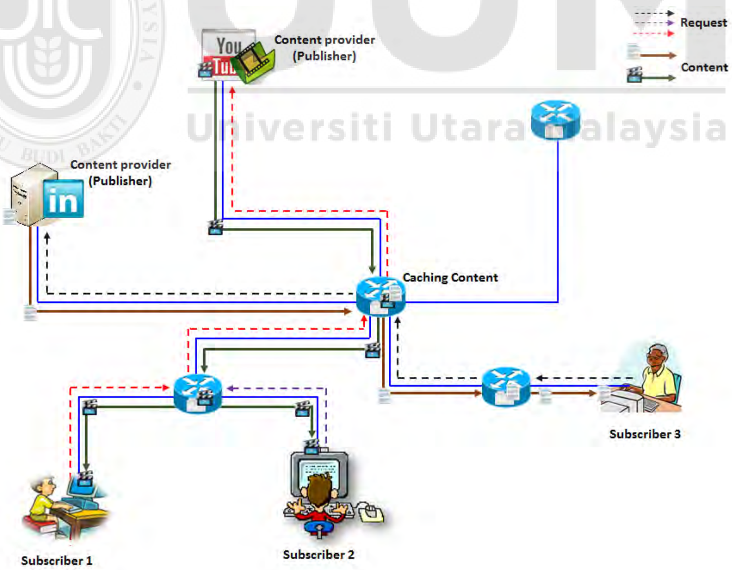


Figure 2.1. Content Caching in ICN

The main important principle attributed to the data-oriented networking particularly in ICN paradigm is the involvement of dynamic way of content caching (see Figure 2.1). This provides fast, scalable and reliable forms of delivering content to attain maximized bandwidth utilization and mitigate congestion. In ICN, an intermediate node (i.e., ICN router) in a traversing path between a publisher and a subscriber has the capability of caching the content objects that traverse its path; this is done in order to create an easier form of data servicing to subsequent requests for the same data objects. Thus, it is a benefiting advantage in ICN as this reduces enormous server visitation for the same content data [26].

Even though ICN is quite a new topic for researchers, several solutions and propositions covering a wide range of various issues under this topic have been achieved so far. However, there are still many challenges and solutions to be developed, and deployment aspects call for more an in-depth investigation in the areas of naming, caching, routing, scalability, management, mobility, and security [1]. Content names are the basic network that are primitive for each architecture proposed to ICN [85]. ICN architectures got to give globally unique, persistence, human- friendly names, security, scalability, global uniqueness, and location-independent [85]. Hence, the major challenge in this section is to develop a naming system which satisfies all these requirements. Practically, there is no such single mechanism or system that fulfills all these needs [1].

Caching is the foundations feature of each architecture proposed to ICN. In reality, the awareness of information enables the network for determining the cached information with no reference to the application layer [86]. The major aim is to improve the performance of the network for content distribution, and improve a better delivery and a more robust transport service. This enhanced performance is gained by reducing the delay experienced by the user and led to an efficient use of bandwidth in the network's core. In addition, reducing traffic near the source of content reduces publisher's processing load. In caching, the main research topics for a network of caches are content replacement policies and content storage policies [1].

Similar to the mechanism of naming, those of routing of ICN have certain desired characteristics as proposed by Bari *et al.* in [85] which are content state, name resolution, retrieval locality, scalability, discovery of closest copy, network-level deployment, discovery guarantee, and security infrastructure. However, there is no such single mechanism that provides all these characteristics. Routing protocols must use techniques to publish packets to all nodes without leading high control overhead that depends on the network size.

Content and subscriber security are more substantial in the ICN design versus the current host-centric design [87]. The ICN architectures security depends on the use of encryption with keys related to the content-name. As ICN is concerned directly with naming content, content properties are still preserved explicitly. However, limited studies have been conducted on the management of these keys, which implies that the responsibility of creating, distributing and revoking the keys lies within the communicator of such information. In addition, the attacker tracks user's information by carefully studying their requests of content when it is available to all NDN routers between the subscriber nodes and the content provider nodes [88].

Dirk, Hannu, and Holger in [30] showed the challenge of mobility in two angles: the name database might be introduced with frequent changes by it, and the content delivery process might get introduced with delays or disruption by it. Bandwidth sharing is not addressed by the majority of ICN. Proposing models which capture the play between bandwidth sharing and in-network caching is crucial, especially for the purpose of caching dimensioning as well as network. The method in which bandwidth has to be shared among contending information flows is even, then an open issue that leads to many challenging questions associated with fairness and efficiency [26, 89].

ICN should be able to serve a very large number of contents (i.e., scalability) [26]. Moreover, the main challenge in ICN is to face the distribution of billions of objects to billions of interconnected devices [51]. According to Bari *et al.* in [85], the largest Border Gateway Protocol (BGP) (e.g., "as discussed by [90]") table contains around $4*10^5$ routes for covering about $3.8*10^9$ IPv4 addresses and $6*10^8$ hosts. On the other side, approximately 10^{12} URLs have been indexed by Google.

In network and in-network management issue, as content object serves generic representation in ICN, it however does not fully refer to the application-level data, rather it could use naming the network entities, such as links, host, and domains or more commonly anything material or intuitive that may be apparent by the reasoning. To juxtapose the extended definition, ICN direction could be projected to capture the knowledge of context information, and thus it may serve as being efficient in designing networks to provide an easy configuration to offer better management [27]. On the other hand, in-network cache approach used for storage in routers is referred to as the management and organization of the contents of the cache by using a different data structure and/or a replacement policy [91]. A data structure is the most suitable for caching data that are recently accessed as contents, thus yielding quick access when the information is needed in order to be traversed again. The replacement policy is efficient for dynamic storage management and the memory extensibility [92].

2.3 Named Data Networking

The various existing ICN initiatives focus on designing an Internet architecture that will replace the current host-centric model and will directly address the problems and limitations identified in the previous section. ICN has acquired the attention of the research community as a new paradigm in networking that can better address users' need in a networked world [93]. A number of designs have emerged in the last few years (see Figure 2.2), including Content-Based Network [94]/Combined Broadcast and Content-Based [95], Data-Oriented Network Architecture (DONA) [32], Network of Information [96], Content-Centric Networking [34], Line Speed Publish/-Subscribe Inter-Networking [97], Named Data Networking [35], Publish-Subscribe Internet Routing Paradigm [98], Publish-Subscribe Internet Technologies [29], Scalable and Adaptive Internet Solutions, COntent Mediator architecture for content-aware nETworks [99], MultiCache [100], Content-Centric Inter-Network [101], Mobility-First [102], and CONVERGENCE [80]. In this research we are focusing and presenting the Named Data Network architecture that it's related with our problem and scope.

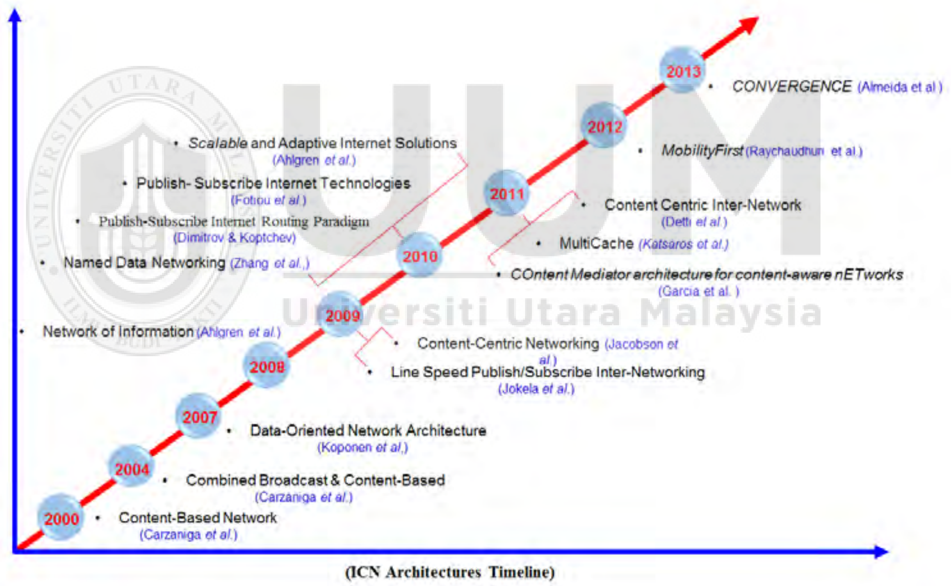


Figure 2.2. ICN Related Architecture Timeline

Named Data Networking (NDN) [35] is a new project under future Internet architecture based on the CCN architecture [34]. The studies activities for this project are based on routing and forwarding improvements for CCN. The goals of NDN architecture are to completely redesign and develop the Internet by replacing IP with content chunks as universal components of transport [85]. NDN is an efficient and simple communication model driven by subscribers who broadcast requests (i.e., Interest packet) to ask for a content by name regardless of the IP addresses of the nodes that supply the content. Interests' packets are forwarded by intermediate nodes (i.e., NDN routers) upstream to publishers that are any node stores or owes the requested content. Publishers simply respond to the Data packets request, which go through the way back to the subscribers [103].

2.4 Classification of Named Data Networking

NDN [42, 43, 31, 35] has attracted much emphasis in the research area recently, with different exploration activities focusing on the rising research with the point of moving from the present Internet engineering which is manufactured and intended for a host-to-host correspondences model. To carefully analyze the NDN architectural design, some important functional characteristics and important auxiliary support are presented in this section as the proposed a taxonomy of NDN (see Figure 2.3). NDN features are classified into three categories: NDN architecture, NDN services, and NDN applications.

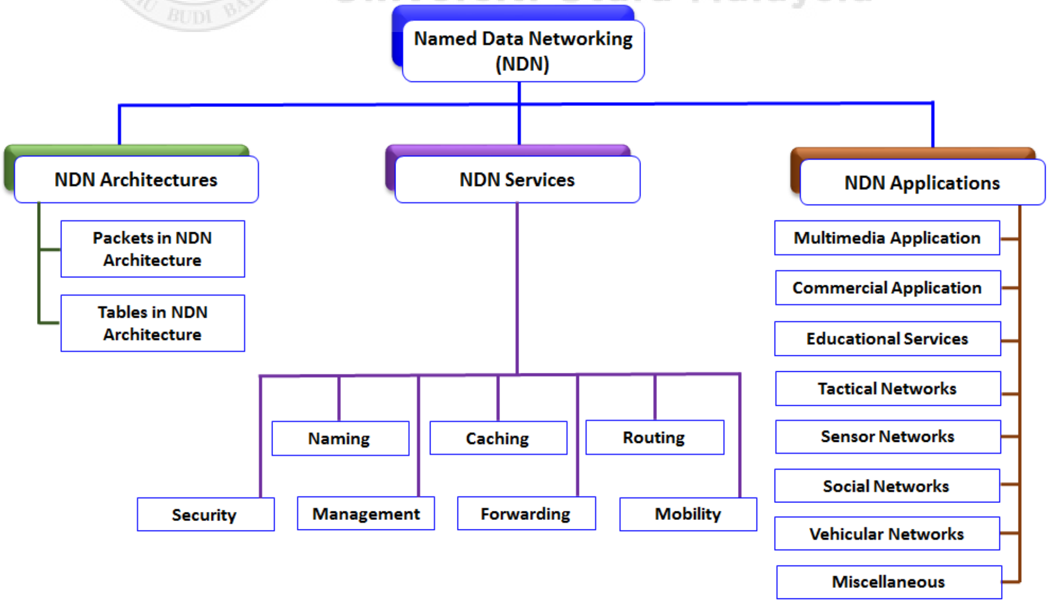


Figure 2.3. Classification of Named Data Networking

2.4.1 NDN Architecture

NDN inherits the IP architecture hourglass shape, but it replaces IP's host-to-host data delivery by a data retrieval model at the hourglass thin waist [42]. This section discusses the organization of different components and their interaction within the system along with its several working principles. Figure 2.4 illustrates the main component of NDN.

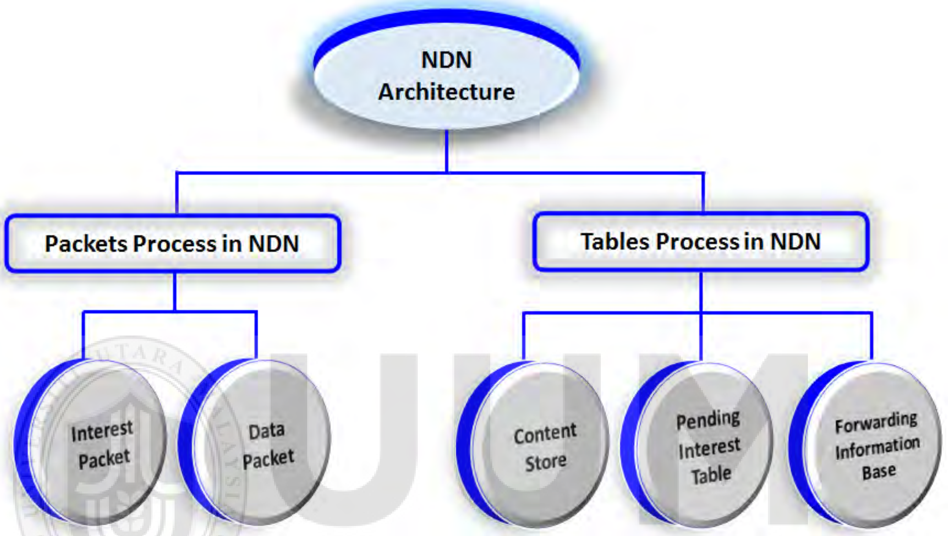


Figure 2.4. NDN System Architecture [35, 34]

2.4.1.1 Packets Types in NDN

NDN architecture introduces two types of packets. The first one is the "Interest" packet as showed in Figure 2.5 (a). By sending these over the outgoing faces, a NDN router announces the subscriber node's demand for content that is named by the Interest packet. Interest packet is simply broadcasted on the available faces in order to get the according Data packet returned by the mechanisms of NDN architecture. These packets usually contain the desired content name by default. Apart from this, the Interest packet is accompanied with selective information such as the scope inside the network, where the Data packet have to come from specific filter information. The Interest packet also has a nonce use for detecting duplicate Interest packets.

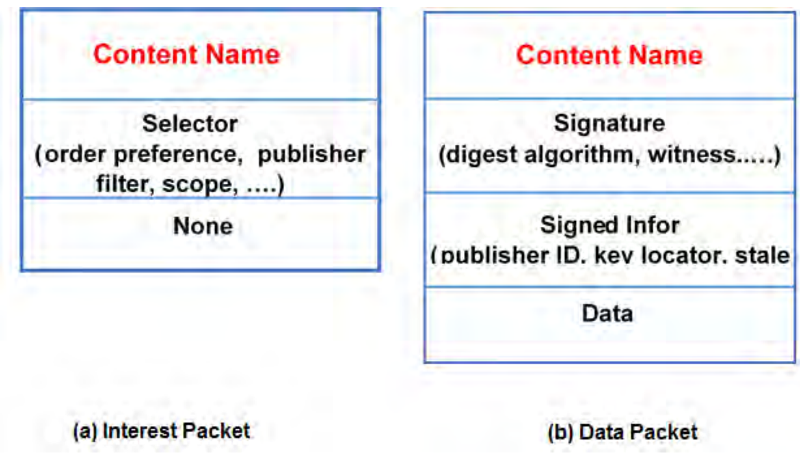


Figure 2.5. NDN Packets Types [35, 34]

The second packet is namely "Data" packet that is used in response to an incoming Interest packet as shown in Figure 2.5 (b). Data packets are considered to satisfy an Interest packet in that they keep a one on one relation, where Interest packet is consumed by Data packet. This rule of thumb keeps a flow balance at each hop and blocks congestion in the middle of a connection path. Content names in NDN are hierarchical structured, this Data packet just serve one Interest packet if its name prefix and the name of Interest packet are matching. Apart from the arbitrary binary data and name, Data packet also has a digital signature that is of certain cryptographic digest of the packet and signed information. This last mentioned field provides extra information regarding the packet including the publisher's ID to locate the key for checking the time stamp or signature. With these ways of verification, it is guaranteed that the packet is identified and authenticated itself, and there is no need of legitimacy by the channel it got transferred [35, 34].

2.4.1.2 Tables Types in NDN

Each NDN router uses three varying structures in processing packet forwarding. These varying structures are the Content Store, Forwarding Information Base and Pending Interest Table [35, 34] as shown in Figure 2.6.

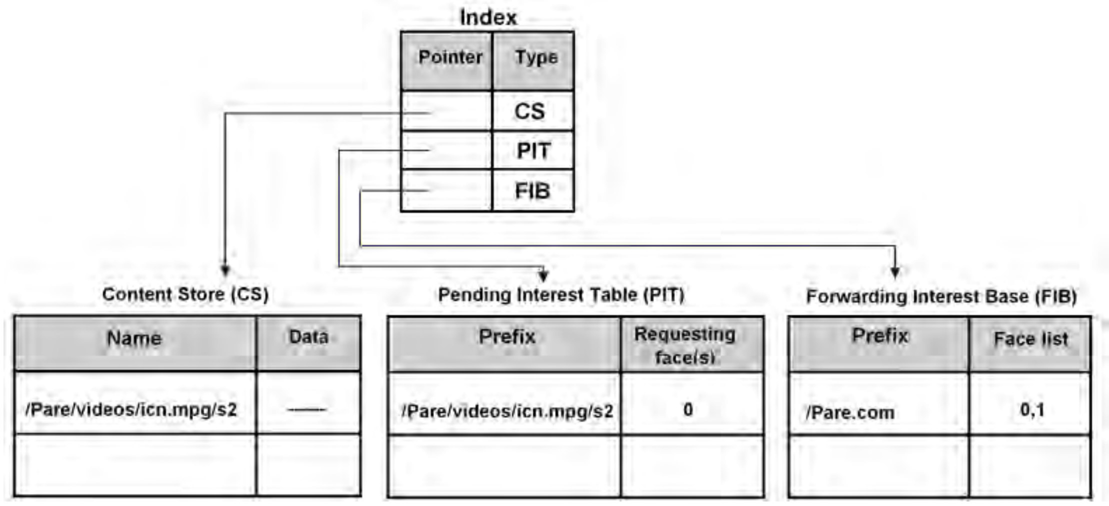


Figure 2.6. NDN Data Structure Tables [35, 34]

- Content Store (CS) is a cache structure (buffer memory) in a NDN router that stores chunks for a very long time by applying for cache update policies. As content is self-authenticating as well as self-identifying, each one of the packets should be useful to certain potential participants in the network nearby. The ability for serving content directly rather than generating additional lookups reduces total bandwidth usage as well as latency.
- Forward Information Base (FIB) is used for storing information on packet forward. It is like a routing table in common IP router. FIB stores information on which faces Interest packets are forwarded upstream towards the source of the content of the question. Hence, the design enables more than one entry that may be needed to be queried in parallel since forwarding is not just limited to one spanning tree.
- Pending Interest Table (PIT) consists of the arrival faces of Interests packet which have been forwarded, but they are still waiting for matching Data packet. This information is needed in order to deliver Data packet to their subscribers. To increase the PIT utilization, the entries of PIT need to be timed out pretty quickly somewhere around Interest packet lifetime.

2.4.1.3 NDN Lookup and Forwarding Routing Scenario

In NDN approach, communications are driven by the data subscribers. To request data, a subscriber sends out the Interest packet that carries a name, which identifies the desired data. For example, the subscriber requests *"www.youtube.com/java.mp4"* (see Figure 2.7). When Interest packet arrives and a content name that matches the requested name is found at a CS in the intermediate nodes (i.e., NDN routers) or publisher's node, the Interest packet is discarded and the content is returned in a Data packet through the incoming face. If there is no match, then the PIT is checked to lookup if it already includes any pending Interest packet in this content's PIT (meaning that this content has already been requested). Then, the NDN router adds the incoming face to this PIT entry list and the Interest packet is discarded. Otherwise, the router remembers the face from which the request comes in, and after which it forwards the Interest packet by looking up the name in its FIB, which is populated by a name based routing protocol.

When the Interest packet reaches the publisher's nodes that have the requested data, Data packet is returned. This Data packet traces in reverse the path created by the Interest packet returning to the subscriber. In this regard, it should be remembered that neither Data nor Interest packets carry any host or face addresses, e.g., IP addresses. Interest packet is sent towards data producers depending on the names carried in the Interest packet, and Data packet is returned depending on the state information set up by the Interest packet at each router hop. In contrast, when Data packet returned, and it arrived at the NDN router, the NDN router is looked up for finding the matching PIT entry and forward it to any or all faces listed in the PIT entry. In addition, NDN routers remove PIT entry and caches the Data packet inside CS table that is basically the NDN routers buffer memory subject to a cache replacement policy. Data packet takes the same path as the Interest packet that solicited it but in the reverse direction. One Data

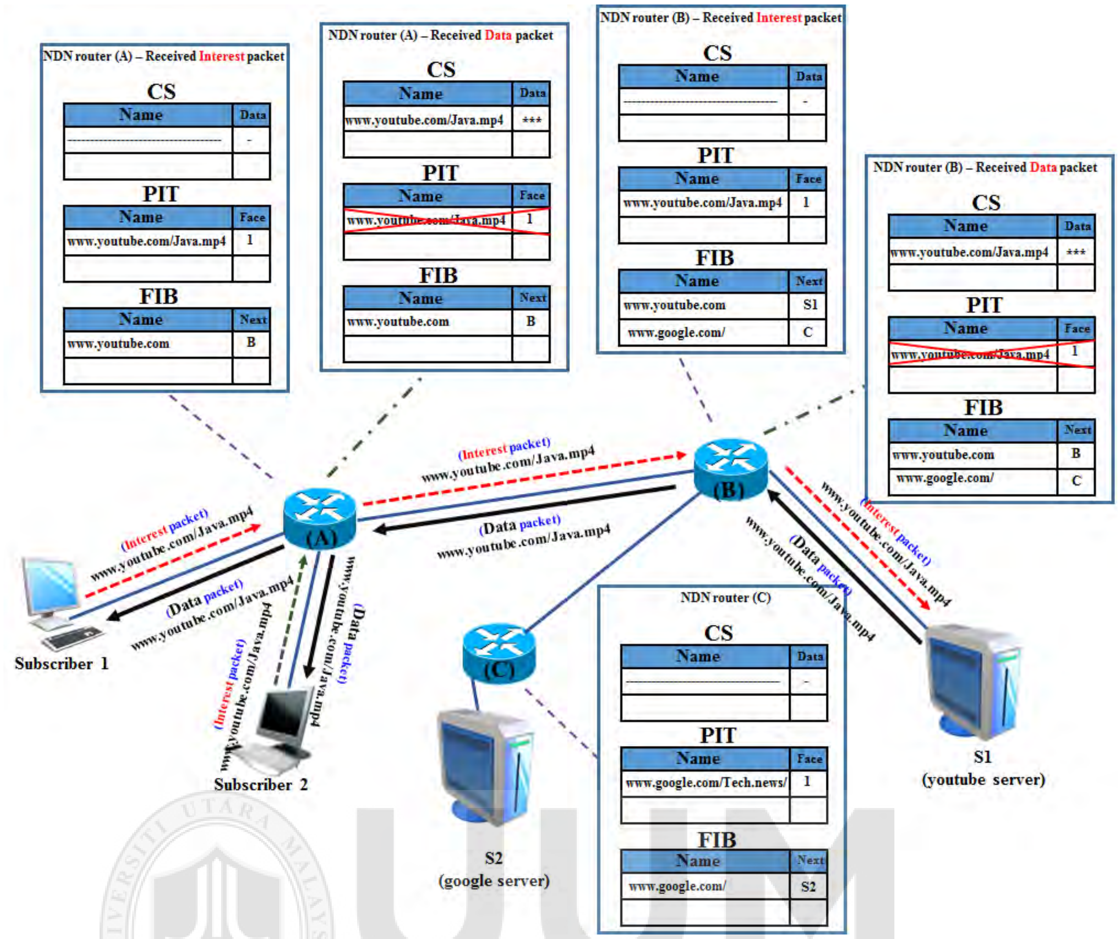


Figure 2.7. NDN Looking up and Forwarding Scenario [84]

packet satisfies one Interest packet over each hop, achieving hop by hop flow balance [104, 35, 34].

2.4.2 NDN Services

NDN services concerns main functional characteristics of NDN, which includes caching, naming, routing, security, management, mobility and forwarding (see Figure 2.8).

Routing: In NDN, routing provides the path and traversing description for name object management in the topology, policies for managing the network lasting changes, and for updating the forwarding table. NDN routing protocol aids in coordinating the functionalities in the forwarding plane that is used in face ranking [105]. The routing

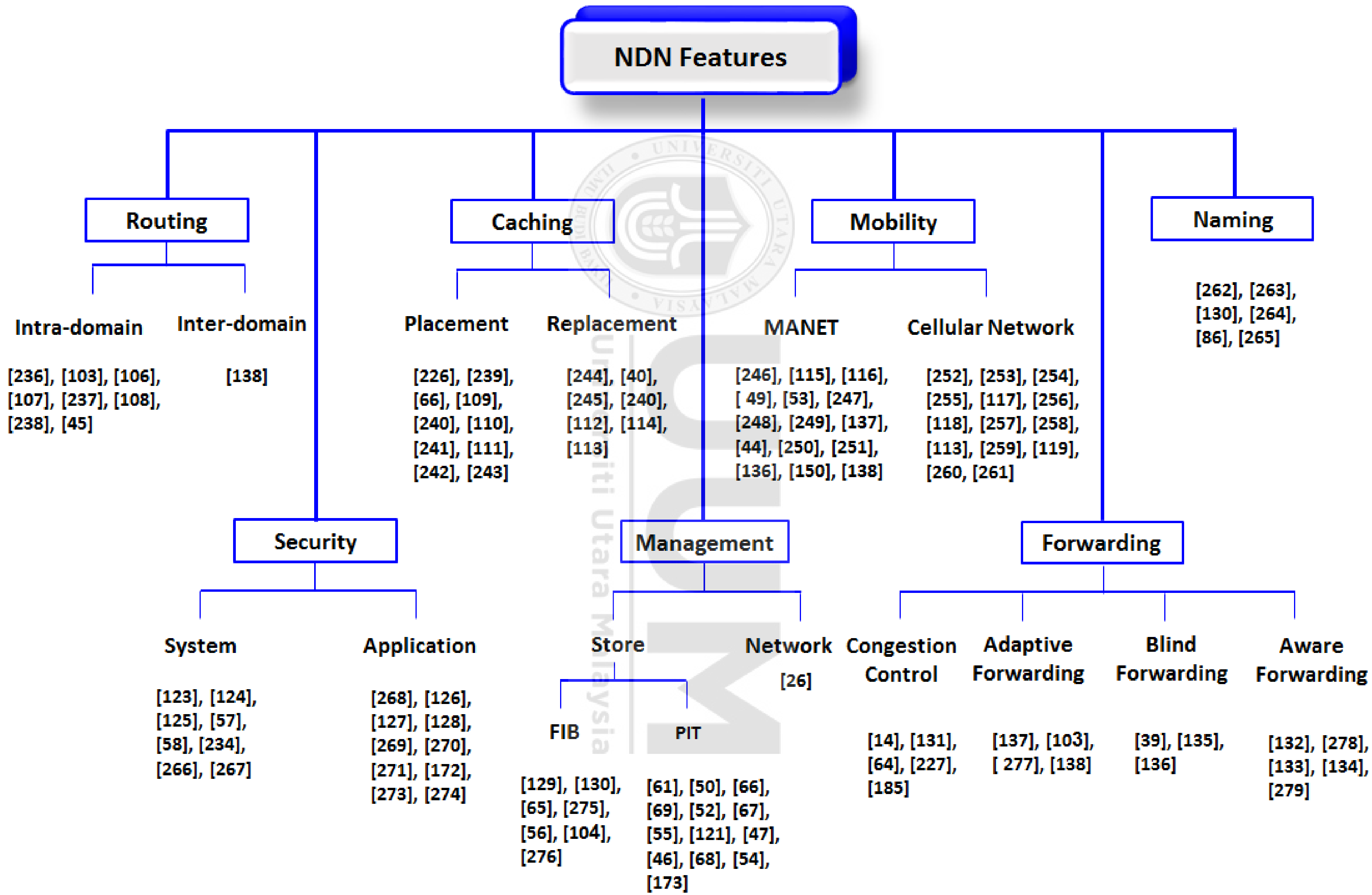


Figure 2.8. Classification of NDN Services

algorithm provides the most suitable dissemination route for Internet, link-state and distance-vector. Then this provides the general routing used in NDN with slight modifications [35]. Routing in NDN can thus be classified based on the domain which is of two types: inter domain (as discussed in [45]) and intra domain (as discussed in [106, 107, 108]).

Caching: Caching in NDN has several benefits due to the wide reachability it provides. Cached contents were implemented with the help of another node (i.e., another NDN router or subscriber nodes) in truly dissociating contents from their publishers. This form of caching in NDN mitigates the overhead at the publishers' side. This is achieved by the provision of multiple copies in the network to save the same contents at several NDN routers in the network. Caching also provides high benefits of dynamic contents for multicast or retransmission when Data packets are dropped or loss [35]. Cache can be classified into cache placement and cache replacement. Cache placement provides the decisions of temporarily saving the content on the network (as discussed in [109, 110, 111]) and cache replacement is the idea of caching the content at the NDN router, particularly when NDN routers' tables are getting to the exhaustion limits (as discussed in [112, 113, 114]).

Mobility: From a broader perspective, the subscriber mobility is a well handled phenomenon due to the subscriber-driven nature of most NDN designs [89]. The subscriber mobility is intrinsically supported in NDN architectures since mobile subscribers can just send new subscriptions of information after a hand-off [38]. In addition, the NDN is thus, in particular beneficial to the Ad-hoc networking environments and other communication medium paradigms. Mobile nodes have the ability to communicate based on the data needed, instead of particularly computing a more specific path to reach a destined node [44]. NDN mobility is further divided into MANET (as discussed in [115, 116, 53]) and Cellular Networks (as discussed in [117, 118, 119]).

Naming: The main abstraction of NDN is the Named Data Object (NDO) which includes web-pages, videos, photos, documents, live streaming and interactive media. Since names are used to identify NDO independently of their location, NDN needs unique names for each NDO [120]. However, naming defines the critical part of all data-centric architecture and applications. Due to the inclusion of the flexibility of dealing with names in NDN, it operates openly on application names, i.e., it merges the network and application layers. Thus, this makes application developers to proffer a solution as to design the data naming model closely in order to guarantee the correctness of application functionality that will leverage all the NDN architectural benefits. The application design is therefore more unique when patterns for Interest packets are expected to be satisfied with different Data packets [121].

Security: In NDN, each Data packet is signed with a traditional public key to guarantee its sufficiency through verification to meet its authenticity [34]. The main requirements of trusted content are origin authentication, data integrity, and relevancy in the data. Publishers use an appropriate algorithm of signatures (like, "RSA as discussed by Rivest *et al.* in [122]") from a huge fixed set in minimizing the size of the verifiable latency of verification, data, and computational cost for data signature generation and verification [34]. The process of signature verification may contain more rounds of certificates used in fetching information and verification. Therefore, the possible solution is to cache validated certificates required for verification which can be used till their expiry. Security is simply classified into system security (as discussed in [123, 124, 125]) and application security (as discussed in [126, 127, 128]).

Management: NDN management is the act of managing in and/or outside resources of the network. It can be classified into two distinct categories: networking and in-network management. Network management (e.g., host, link, domain, and topology) or more generally is anything material that may be perceived by the senses. In other words, it is the process of administering and managing the networks in order to efficiently design networks with ease of configuration for management and for general maintaining the quality of service. In-network management, NDN describes the form of cache operations as related to the position, the size, time and manner, in which contents are placed, replaced and removed from NDN routers in order to grant the ability of making information readily and easily accessible. Moreover, In-network management is further classified into PIT (as discussed in [50, 52, 55]) and FIB (as discussed in [129, 130, 65]).

Forwarding: NDN forwarding plane is built with the ability of detecting failures (from links, node, and packets). For performing recovery actions, routing needs not be performed continuously on FIB updates to improve the scalability and stability of the NDN routing plane [37]. For the purpose of this research, NDN router forwarding is classified into forwarding strategies and scalable forwarding. Scalable forwarding supports intelligent and stateful traversing. Forwarding plane in NDN router also acts as a control plane due to its ability in per taking in forwarding strategies in all the decisions needed for the Interest and Data packet forwarding. Forwarding strategy in NDN router determines how to use multiple forwarding options efficiently and choose the best face(s) to forward the Interest packet. Among other forwarding strategies deployed by the NDN router forwarding plane are congestion control (as discussed in [14, 131, 64]), adaptive forwarding (as discussed in [132, 133, 134]), blind forwarding (as discussed in [39, 135, 136]), and aware forwarding (as discussed in [137, 103, 138]).

2.4.3 NDN Applications

NDN provides the practical benefits of well-designed named-network architectures in real life through different traditional as well as novel applications. According to the syntax and semantics of the underlying NDN architectural design, NDN supports naming features in its development for a robust network and applications. NDN naming is purely an application-specific concept. Hence, this section shows various naming schemes in existing NDN prose alongside several applications as illustrated in Figure 2.9.

Teleconferencing is a standout emergence amongst the most prominent multimedia application, e.g., video-gathering, video telephone and audio chatting, tele-presentation, tele-educating, and tele-musical practice. A conveyed, server-less, multi-client visit application is actualized over NDN (see [139, 140, 141]). In Commercial applications, the primary administrations upheld under these applications are electronic installments from anyplace, steady support among various datasets, document facilitating administrations for record synchronization, vehicle correspondence, for example, information accumulation from vehicles, information gathering, cases found in news, about road conditions, accident data, climate upgrades among vehicles (see [142, 143, 71]). From the educational perspective, the primary point of this service is to present the essential system design advancement over the time. Essential applications are to set up the instructive systems like conference rooms and to provide ad-hoc correspondences amid conferences, meetings, and seminar (see [144, 145, 146]).

In addition, a Wireless Sensor Network (WSN) is made out of sensor nodes, which are densely deployed, inclined to failures and disappointments, and are asset compelled. WSN works in an information driven fashion as queries are addressed with the required information. A portion of the NDN benefits over WSN (as discussed by Amadeo *et*

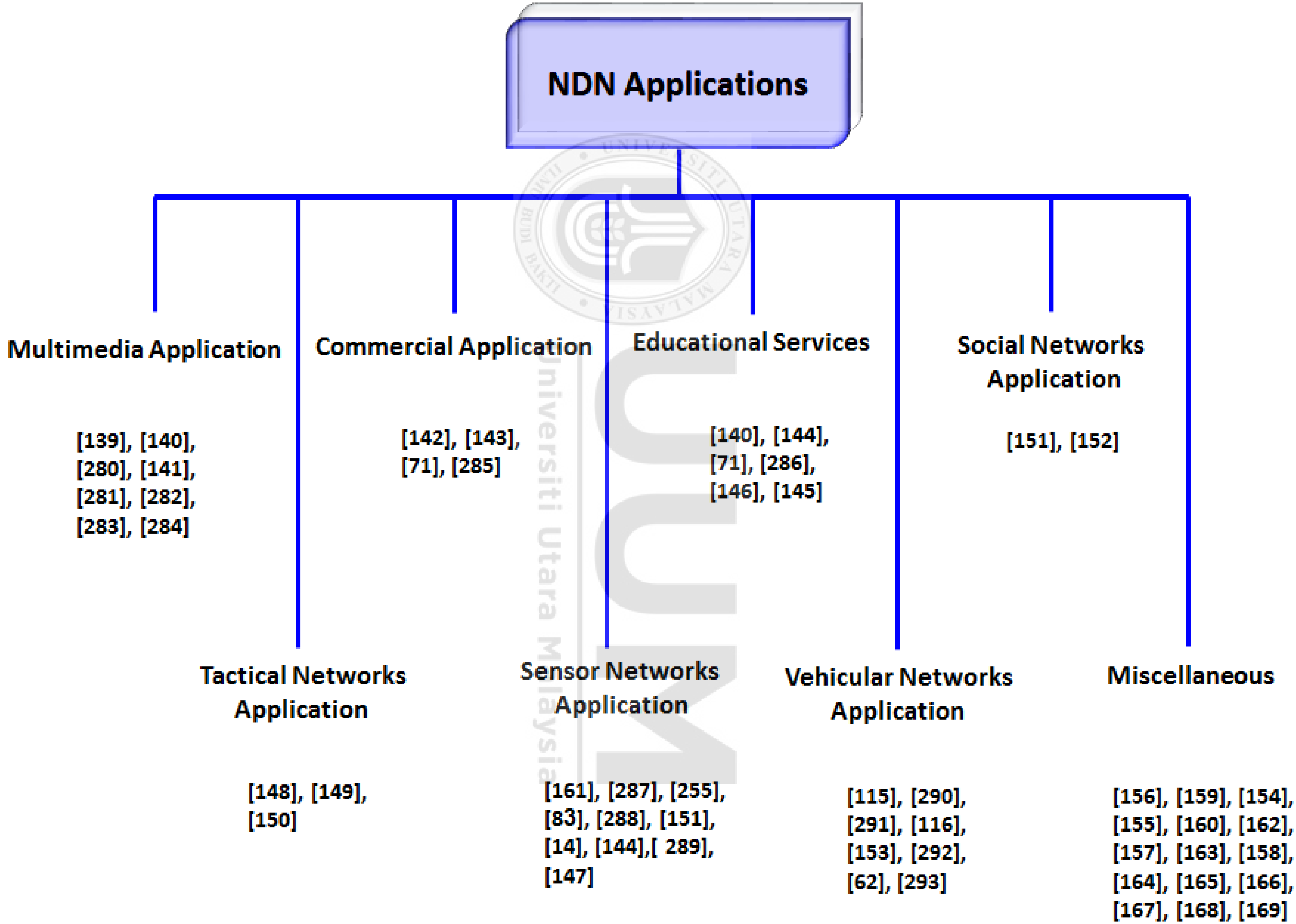


Figure 2.9. Classification of NDN Application [38]

al. in [11]) through quick information recovery as NDN uses various leveled naming, scalability as NDN inalienably bolsters broadcast correspondence, caching, simple application improvement, and adaptable arrangement (see [38, 83, 147]). Tactical networks are an ad-hoc communication network which connects military devices and soldiers in the combat zone with the military headquarters in connection to the army bases. A typical, tactical or calamity recuperation situation is naturally of numerous levels set in two ways-nodes move in cluster-like (emergency rescue team or coalition operations) and they are associated through network gateways and backbone. For this reason, data have nearby pertinence (most information is made and sorted locally). There are both mobile nodes (e.g., agents, soldiers, mobile sensors) and stationary nodes (e.g., a command center, fixed sensors) (see [148, 149, 150]).

Consequently, social networking exponentially grew in the last few years. Nowadays, millions of users interact with each other through Facebook, Twitter, Digg, Google Plus, and other social media platforms. Numerous forms of Web-based applications have been submitted to be driven in a content-centric approach. In the traditional web usage, Users/subscribers issue demands through requests for specific bits of contents, with totally neglecting where the requested content is set in the system, which is a regular "content-centrism". Additionally, web advancements have been utilized to bolster both Online Social Networks and User Generated Content applications, which present the cases of content-driven applications. This kind of utilizations is normally upheld and supported by NDN (see [151, 152]).

Moreover, vehicular communication in network systems has been a critical consideration in the most recent couple of years through advancements, particularly with the advent of the NDN concepts. Scientists and researchers are quick to examine the attainability of utilizing the content-driven approach in dealing with the enhanced productivity in vehicular system networks. Recently, vehicular NDN has been massively developing for effectively giving more portability support in the foundation or infrastructure-less systems. Consequently, NDN has been adapted in VANETs (Vehicular NDNs (VNDN)) by researchers of the network domain [115]. Along these lines, driving the communication paradigm from host-based system into the NDN for vehicular communications (see [115, 153, 62]).

Several studies have identified the use of NDN in different domains, such as developing communication APIs [154, 155], gaming [156, 157, 158], Climate Research and HEP [159], Inter-Process Communication [160], smart city [161, 162], home and enterprise networking [158], entertainment [163], multi-sensor collaborative sensing [164], emergency communication support during disaster [165], handling malicious nodes [166], traffic control for power saving [167, 168], instant messaging [169].

2.5 Pending Interest Table

Pending Interest Table (PIT) is the main component in the context of NDN router design. It is one of the three principal special data structures recently inculcated into the NDN router designs in order to attain full functioning of the NDN architecture by also retaining some of the current Internet semantics [170]. Each PIT section has five fields [47] as delineated in Figure 2.10 including name associated with the entry (content name), list of incoming face(s), list of outgoing face(s), time when the entry should expire, and any other forwarding strategy.

Name	List of Incoming Face(s)		List of Outgoing Face(s)			Lifetime	Forwarding Strategy
Content name	Interface ID	Arrival time	Interface ID	Send time	Retransmission Number	Entry expire time	Strategy name

Figure 2.10. PIT Entry Fields [47]

Content name is uniquely identifies an Interest packet, which prevents the replica packet sending (known as loop detection). The <<face ID, arrive time>> tuple represents a full list of incoming faces of the requested information from the subscriber with the related added information (such as entry time of the Interest packets to the face). The <<face ID, send time, number of retransmission>> tuple implies a list of the active faces of which the Interest packets have been sent with its related data (e.g., time at which the Interest packet was sent on the face, number of retransmission of the Interest packets on the face). The lifetime represents the time, whereas a data entry terminates (i.e., the maximum lifetime among all the got Interest packets for the same name).

The PIT in NDN router does not only keep track of the Interest packets that requires to be forwarded for upstreaming to a data source, but it saves the face in which the Interest packet is received. At every point of data traversal, an entry is saved and recorded to the PIT. The cached or saved Interest packet is in turn deleted upon upstream data fulfillment through delivery or when the PIT entry lifetime expires. An important aspect of the PIT forwarding is the fact that when multiple requests amount to the same Interest packet, Interest packet names are aggregated which saves duplicating forwarding. In this case, only the instances of the same Interest packet are stored. The resulting saved faces are further seen as a help to instigate the forwarding of the returned data to the request subscribers [170].

2.5.1 PIT Significant Features

According to the Dai *et al.* in [46], PIT has some significant features which are:

i. The enabling of the PIT and Data packets to be traversed through routing without necessarily specifying the source or destination. This implies that the PIT has the ability to make NDN communications with higher Interest packets on the content rather than the location of the contents or the content exchanger.

ii. The feature above inherently supports communications anonymously, making network attacks hard to dispatch and correspondence engagement more secure.

iii. In PIT, loops are persistently prevented due to named Interest packet forming a random nonce that is stored as the PIT entry. When this is placed, duplicate faces are effectively discarded. Therefore, Data packets are prevented from looping since reversed paths are used for traversing Interest packets.

iv. The property above also helps NDN router in inherently supporting routing using multi-path. This is also due to the NDN router that has the capacity of sending out Interest packets using multiple faces with less emphasis to loops.

v. PIT features include the provision of the support for Data packet multicasting when multiple Interest packets are received through NDN routers that have the same content.

vi. PIT feature also includes the detection of Data packet loss when Interest packets are to be responded to or beyond a time threshold.

All the above aforementioned critical elements of PIT in NDN routers empower the NDN architecture to be an information/content-driven network. In addition, with the current fast packet arrival rate, PIT will have a huge size, and thus it requests a greater degree in accessing information (deletion, insertion and lookup) frequency. These have brought about a wide level of headed research on the attainability of PIT and content-centric through exploration. Components that PIT conveys make it vital and fundamental in NDN router. Hence, PIT content management remains a knotty obstacle which hinders the practical implementation as well as development of NDN.

2.5.2 PIT and Research Issues

There are yet many challenges and solutions that need to be proffered in the development and deployment aspects of PIT. Thus, this calls for in-depth investigation in terms of PIT since it has become one of the most significant and highly important research area in the field of content-centric. Raaid *et. al.,* in [171] highlights some open issues in PIT, which is identified and desirable properties for PIT.

i. ***Lookup operates***: since the PIT increases when subscribers sending their Interest packet and decreasing when Data packets arrive at the NDN routers. In case of the number of PIT entries increases, these required access speed to PIT structure and the possibilities available in this meaning by using current memory techniques, that make the PIT size representable as the main bottleneck that affects the whole NDN infrastructure. Because of the PIT is very dynamic for all incoming Interest packets and matching Data packets, a special process (i.e., lookup process) must be happening in the PIT which should to be performed faster. This requires fast memory that is unfortunately merely offered for small storage sizes (see [55, 56]) .

ii. ***Filtering***: one of these issues, malicious users can create artificial Interest packets in order of filling the PIT on NDN routers. Hence, it is implementing a Distributed Denial of Service (DDoS) attack. This type of attack can possibly be implemented by distributing the generated request packets that include valid destination prefixes without existing resource names, in order that the NDN routers correctly forward Interest packets and keep new entries inside the table. Nonetheless, replies never come back. Another issue of security in NDN architectures is the vulnerability of NDN router to the PIT pollution attacks. This type of attack includes sending random Interest packets for content as a way to modify large values for the lifetime field. Thereby forcing NDN routers, for storing unpopular entries in their PIT (see [57, 58]).

iii. ***Entries situation***: PIT should provide low-routing overhead, metadata updates, avoids congestion, low-latency entry operations addition and deletion. The interesting question at this point, how to determine that the Interest packet deletion must be expiry-time based on some hybrid or explicit schemes.

iv. ***Interest Lifetime***: other issues are, when the dropping of both the Data and Interest packets occur on the way as a result of the network impairments, deciding where the NDN router or the endpoint could be, setting the timeout in duration and retransmitting the Interest packet after a timeout remain the key issues that could affect the performance of the entire network [170]. In addition, the impact is noticed directly on the occupancy of the PIT. According to the CCNx protocol technical documentation, "*A node MUST retransmit interest packets periodically for pending PIT entries*" [172]. However there is no unanimous agreement on how often should this periodic retransmission be done.

v. ***PIT overflow***: the PIT is very dynamic due to the high speed packet arrive rate to the PIT. Thus, for all incoming request packet and matching Data packet, hence a special process must be happening in this table. These processes should have to be performed faster to avoid this table may be overflowed, which cause the delay and packet loss for these packets because the PIT receive and remove the Interest packets exponentially. It is not easy to predict the tables are full. If the table is overflowing, subscribers' requests will be discarded from the NDN routers, and based on this, subscribers will experience an increasing Interest retransmitting rate that will lead to a complete collapse of the whole network (see [59, 60, 58]).

vi. ***PIT replacement***: cache memory remains one of the most studied topics in the computer architecture research. With the ever-increasing speed added between processor and memory, this emphasizes the need for a more efficient memory hierarchy for better network and content manageability. When PIT sets become full and a new Interest packet of memory needs to be placed into the PIT, there should be a provision for replacement based on the built-in algorithm for content placement. Replacement policy, therefore is one of the most crucial factors of determining the effectiveness of a cache memory. Its importance becomes even more essential with the latest technological trends moving toward a highly associative cache support paradigms [76]. Consequently, the most crucial issue among them is the PIT management i.e., to specify what Interest packet is to be replaced and what is the cost in replacement in an event that the table overflows, with subsequent service disruption and possible network collapse.

2.6 PIT Management Techniques

The state of PIT at each NDN router has several critical functions. First, PIT includes the number of faces through which Interest packet has arrived. In addition, it provides natural support for multicast functionality. Second, the router can control the rate of

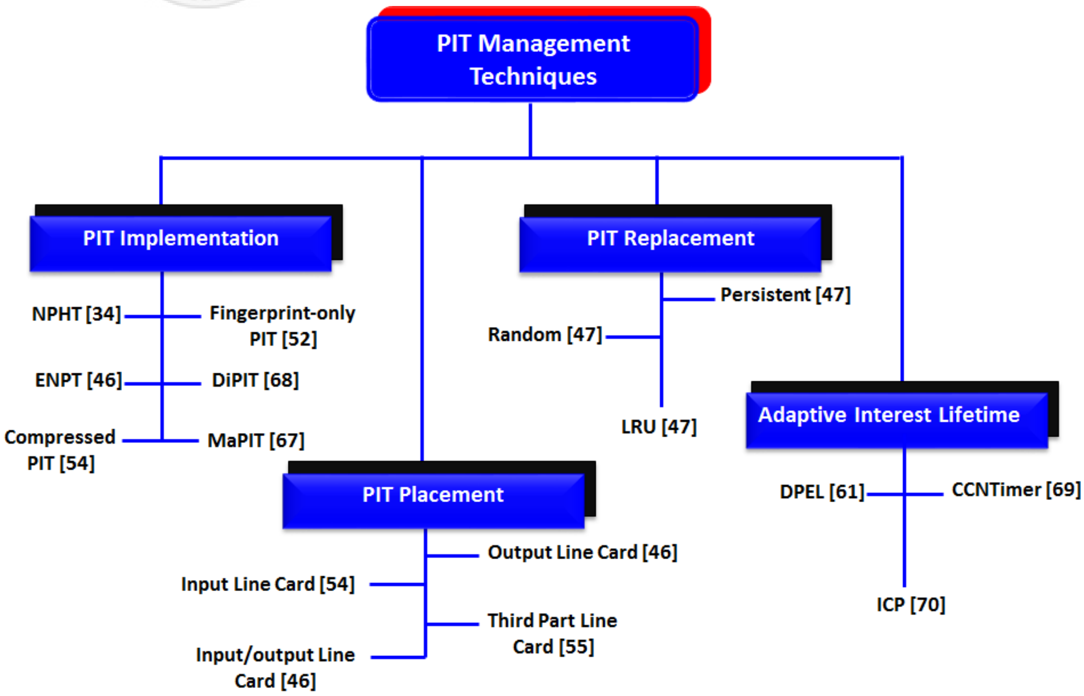


Figure 2.11. PIT Management Techniques

incoming Data packets by controlling its PIT size. Finally, PIT with data caching, an NDN router removes the dependency on transport protocols to prevent congestion collapse [35]. Based on that, enhancing the management operations of PIT is required. In this study, PIT management techniques classify into four categories: PIT implementation, PIT placement, PIT replacement, and adaptive Interest packet lifetime. Figure 2.11 shown the main PIT management techniques based on previous studies.

2.6.1 PIT Implementation Techniques

The PIT may present stringent restrictions in terms of scalability. The challenging task is the design of a scalable PIT because it needs efficient data structure to implement PIT since it requires per packet updating and longer naming. This increase in the size of PIT leads to memory consumption, lookup delay and decreases the quality of PIT. To address the problem, several PIT approaches have been proposed, which are classified into three categories: Hash Table (e.g., NPHT and Finger print-only PIT), Trie Structure (e.g., ENPT), and Bloom Filter (e.g., DiPIT, Compressed PIT and MaPIT).

2.6.1.1 Name Prefix Hash Table (NPHT)

NPHT [173, 34] is the first and simple approach in which PIT and FIB share the hash table logical. Two kinds of structures are indexed, which are the Forwarding Information Entries (FIEs) and the Propagation Entries (PEs), where metadata about pending Interest packet info and forwarding info are kept respectively. Each Interest packet in NPHT points to PEs and FIEs. In PIT, Interest packets are presented in form of PEs. Each Interest packet has a unique nonce field. Propagating Hash Table (PHT) is established to store all the nonce fields. Because of the uniqueness propriety of nonce fields that is unique for each Interest packet. PHT stores all the nonce field of the Interest packets presented in PIT. The benefit of this design is to prevent loops in the network due to its impact on the operational flows.

2.6.1.2 Fingerprint-only PIT

Fingerprint-only PIT [52] proposed an PIT plan that utilizes fixed-length fingerprints rather than name strings. Fingerprint-only PIT promises Interest packet delivery with a minimized representation of the storage. This design approach is based on ideas that storing fingerprints save memory space, and that edge NDN router can aggregate most of the duplicate Interest packets. Subsequently, this system-wide solution for versatile PIT can unwind the Interest packet collection necessity for the core NDN routers. Interest packets are accumulated at the edge switches and afterward arrival at the central network. The central NDN router essentially forwards the overall arrived Interest packets. Finally, the main Interest packet bundles are gathered at the edge NDN routers before coming to the publisher node. The publisher node receives only one Interest packet request. At that point, one Data packet is replayed and distributed to the subscribers. The entire packet processing procedure is transparent to the subscribers and publishers. The PITs in edge NDN routers operate as presented in the original NDN design, where name strings are stored, and Interest packet aggregation is sustained.

2.6.1.3 Encoded Name Prefix Trie (ENPT)

This approach denoted by ENPT [46] relies on encoding the Name Component Encoding (NCE) approach to shrink the size of PIT by speeding up the performance of lookup, insertion and deletion operations. NCE stores the elements as character strings to be searched within the table. A Name Prefix Trie (NPT) strategy is utilized to emerge the names with the same prefix. Because of the nature of NDN, the names will be organized in a hierarchical way. This contributes to more organized and manageable PIT. However, shrinking the PIT size in this way is not sufficient for faster PIT access. Therefore, the names are associated with codes (integer values), thus the PIT operations, such as lookup, deletion, and insertion, will be faster.

2.6.1.4 Distributed Bloom Filter based PIT (DiPIT)

The approach in [68] implementation of PIT is denoted by DiPIT. This approach performs better on PIT in terms of speed, space, and cost of content retrieval. For DiPIT, Bloom Filters (BFs) are an important part because it is faster updated and lookup operations. In addition, these are very effective for memory utilization, but the ratio of "false positive" can be increased in this regard. DiPIT-based design builds one PITi (i.e., a small PIT table) on each NDN face. In the DiPIT design, BFs are used to construct PITi tables. Moreover, all faces share an additional BF. The idea behind this design is that each PITi works independently and the footprints of the Interest packets are recorded in a Counting Bloom Filter (CBF) that arrive from the perspective face. In this way, Data packets that attained all PITi tables are parallel, checked and then forwarded on the faces if there is any matching footprint in the PITi.

2.6.1.5 Compressed PIT

Compressed PIT approach [54] is a proposed PIT compression with United Bloom Filter (UBF) to compress PIT effectively. For compressing PIT with Bloom Filters (BFs), the key is a face as well as the value is a name set. In the transposed PIT entries, it is associated to each face, and there is one BF that exemplifies the name set. The essential form of BF is not suitable for this compression because the deletion of elements is not supported. However, BF is utilized to represent each one of the possible extensions of BF. For each received Interest packet, CCN router checks the Interest packet name; if it is matched in any of the Bfs, the Interest packet will not be forwarded. On the other hand, for each Data packet, which is received from a face, NDN router will check the Interest packet name in all available BFs. The Interest packets will be forwarded to the faces with BFs that contain the name. UBF is proposed to get rid of the need for deletion in order to avoid such unexpected anomalies.

2.6.1.6 MaPIT

To enhance the implementation of PIT to fulfill the existing network requirements and memory technology, Li *at el.* in [67] proposed a new data structure Mapping Bloom Filter (MBF) as well as an improved PIT called MaPIT to meet the PIT requirements and make its usage of the existing memory chip. MBF is a data structure improvement of BF to support mapping and querying the set elements in the memory as well as lowering on-chip memory utilization. MBF is comprised of two modules; the first one is the Index Table (IT) and the second one is a Packet Store (PS). IT is deployed on the on-chip memory to get into the PS. Recorded contents will be designated for dispersed locations on the PS that is fixed cache in the off-chip memory. The Index Table consists of two structures, i.e., a regular Bloom Filter (BF) and Mapping Array (MA). While using the BF, the elements are checked if they are available in the MBF or not. To reach the PS, the MA value is utilized as the offset address of the PS. MaPIT consists of two storages, which are off-chip memory and an on-chip memory. The off-chip is the memory, where CBF is working, whereas on-chip memory is the one, where the IT of MBF is working as a summary. Table 2.1 presents a comparison to clarify the whole idea on the PIT implementation approaches in NDN in terms of technique name, evaluation type, environment used, performance metrics, focus on and drawback.

A comparative Table 2.1 presented to complete all idea on the PIT implementation approaches in NDN. NPHT approach is effective during the normal operation because the main idea is to implement PIT. The hash table is capable of doing a quick lookup operation, but it has the following drawbacks. Because of the architecture of NDN,

Table 2.1
PIT Implementation Techniques Summaries

Technique	Environment	Performance	Contributions	Drawback
NPHT [173, 34]	Testbed	Memory usage	i) Implementing FIB and PIT by sharing the hash table logically. ii) Easy to implement. iii) Quick lookup operation.	i) Increasing the delay time. ii) Not efficient for scalability issue. iii) Bits number per element for the storage is quite high.
Fingerprint-only PIT [52]	Simulation	Memory usage, overhead, latency	i) Enhance the implementation of PIT. ii) Reducing the PIT size. iii) Overcome duplication Interest, fingerprint collisions.	i) Poorly explanation of lookup when the link speeds increase. ii) Not include operation to avoid PIT overflow.
ENPT [46]	Emulation	Memory usage, PIT access frequency speedup	i) Relies on encoding the NCE to shrink the PIT. ii) Speeding up the performance of looking up, insert and delete. iii) Addressing scalability and improving PIT.	i) Needing more complex architecture to achieve shrink PIT. ii) Require a special encoding algorithm. iii) Consuming much more storage space.
DiPIT [68]	Real test in last CCNx	Memory usage, size, throughput,	i) Splitting several sub-PIT on every face of router. ii) Increasing throughput for an incoming packet. iii) Reduce the memory usage.	i) Couldn't filter Interest duplications. ii) Impacting CBF when the false negative is occurred. iii) Entry lookup should check on every PITi.
Comperssed PIT [54]	Simulation	Memory usage	i) A new extension of BF, to compress PIT effectively. ii) In UBF, the compactness may be organized because there is no incorrect deletion.	i) Increasing routing overhead. ii) Not mention how to avoid PIT overflow. iii) Interest with similar name can be forwarded twice.
MaPIT [67]	Testbed	Memory usage, probability of false positive	i) Enhancing the implementation of PIT. ii) Minimizing the on-chip memory consumption. iii) Efficiently in time and space.	i) Proceeding algorithm requires an additional much more storage space. ii) Not include operation to avoid PIT overflow.

the PIT depends on rigorous matching, where each record must be remembered; this requires more memory space. This situation becomes worse if the number of entries has increased because it results in increasing the delay time experienced by a subscriber. In such a case, the NPHT approach will be not efficient for scalability issue. In addition, there is no clear operation to avoid PIT overflow; therefore, this can affect memory usage and increase the packet loss. Fingerprint-only PIT approach is implemented to reduce the PIT table for a core router with SRAM/RLDRAM chips, and it overcomes duplication Interest packet request and fingerprint collisions. However, this approach has given a poor explanation in the side of lookup when the link speeds increase. Moreover, the approach does not include an operation to avoid PIT overflow. Hence, this can affect memory usage and increase the packet loss in both core and edge NDN routers.

ENPT obtains some benefits from both accelerate the access throughput of PIT and reduce PIT size, thus resulting in the decrease of the memory utilization. Although this approach improves PIT by arranging the PIT entries in order to ensure high insertion, deletion, and search speed as well as shrinking PIT size. However, this approach needs more complex architecture to achieve these objectives [58]. Since an additional special encoding algorithm has been required, much more storage space is consumed. DiPIT approach is an efficient PIT approach based on Bloom Filter. It can increase throughput for an incoming Interest packet to reduce the memory usage (up to 63%) in the NDN router. However, this approach does not filter Interest packet duplications since the same Interest packet arriving at different faces are recorded individually at different PITi. Moreover, this technique is not mention the impact of CBF at PIT when the false negative occurs if any of the counters decrease to zero, thus leading to Interest packet loss and redundant traffic. Furthermore, this approach did not mention what should happen if PIT becomes an overflow that can affect the performance approach.

Compress PIT with Bloom Filter (BF) approach uses an extension of BF, namely UBF to eradicate the demand of deletion to avoid such unexpected anomalies. Hence, the communication can be easily recovered from the errors through retransmission. The most important advantage of UBF is that the density can be controlled and there is no incorrect deletion. This approach can reduce storage space required by about (40%) with only (0.1%) error probability. However, the drawback of this approach is increasing the routing overhead because Interest packets which have the same name is forwarded twice or more times in one epoch couple and this approach does not include an operation to avoid PIT overflow. MaPIT approach is presented to enhance the implementation of PIT. This approach can minimize the consumption of memory to (2.097 Mbytes), lowering the probability of false positive to be under (1%) for (2000000 names). In addition, it is implemented on SRAM memory as on-chip memory to satisfy today's memory technology as well as network requirements. Hence, the proceeding algorithm has required an additional much storage space since several additional components are included in this approach. Moreover, it is not mentioned how to avoid PIT overflow when it occurs.

2.6.2 PIT Placement Techniques

Although PIT data structures reduce memory consumption during high link rates, the memory requirements can exceed the size of a single memory chip. PIT placement refers to where in an NDN router the PIT must be implemented [55]. Therefore, there is a need to divide the PIT; one is for each incoming/outgoing face of the NDN router to handle subscriber requests at the line speed [38]. Consequently, there are four placements to handle this issue: input line, output line, input/output lines and the third part line card.

2.6.2.1 Input Line Card Placement

Input line card placement [68, 54] is indicated that a PIT should be composed in each input line-card. Accordingly, a requested or sent Interest packet creates a PIT entry only in the PIT of the line-card when received. In the event that the corresponding Data packet is received at the output line-card, the Data packet are broadcasted to all input line-cards for the PIT which further indicates a decision of the Data packet being forwarded or otherwise.

2.6.2.2 Output Line-Card Placement

As proposed in [46], an optimal positioning of the PIT should be on each output line-card. Thus, an Interest packet should not necessarily create the PIT entry when it is received in the input line-card, rather it should use the output line-card forwarding, and the selection should be done using the LPM of the FIB. By using the approach, the aggregation of Interest packets is received at several line-cards, which also proves some multi-path cases limitations. The scenario works in the fashion of receiving Interest packets that are handled at a line-card k being forwarded to two distinct output line-cards, l and h. Upon satisfaction, the Data packet returning route is used through forwarding the line-card k. Accordingly, the resulting Interest packet creates two entries in PIT_l and PIT_h, respectively. The challenge therefore remains that the line-card l and k is unable to detect if Data packet were received earlier at the other line-card. In addition, it is assumed that another Interest packet is received at line-card k and is sent to line-card l with another Interest packet being a different Interest packet from the initial, for the same Data packet at line-card k but is sent to line-card h. This implies that the initial received Data packet on line-card l or h, should satisfy both Interest packets in practice; therefore, two Data packet are needed for this PIT's placement.

2.6.2.3 Input/output Line-Card Placement

This placement was initially proposed by the Dai *et al.* in [46], but it was dismissed in favor of the output-only placement. This implies that it is important to place the PIT in both input and output line-card positions. Similarly, Interest packet makes a PIT entry for both the input line-card at the point that it is received and at the output line-card, where it ought to be sent. In contrast with the output placement, this placement has two advantages: the absence of the unnecessary look-ups in the FIB and duplicated Interest packets in presence of multi-path.

2.6.2.4 Third Part Line Card Placement

This placement was originally discussed in [55], which indicates that a PIT should be placed at each input line-card as in the input-only placement. When an Interest packet is posed for content C at a line-card, it is "*delegated*" to a third party line-card as the name implies. The resulting third party line-card is therefore selected as the $k = content_{ID} \, mod \, N$, since N is the number of line-cards in the NDN router and $content_{ID}$ is the hash (e.g., CRC-32) that bears the content name or $H(C)$. Consequently, the content aggregation is done for all entries of C to the PIT completely independent on the line-card that bears the Interest packet received for C. From the output point, no PIT is required. When the Data packet is acknowledged through receipt, the output line-card identifies K through the operations of a modulo evaluation as: $H(C) \, mod \, N$. Executing operations and other actions enable the placement for both multipath and loop detection as the third party line-card to act as the entry aggregation point. For instance, when two Data packet are expected at two different output line-cards, the initial received Data packet are forwarded to a third party line-card, where it subscribers each pending Interest packet. The same dissemination pattern is followed at the instance of the Data packet received as the second part. It is then forwarded to a third party line-card. Table 2.2 summaries the characteristics of all the line card placement discussed in this sections.

As the PIT keeps track of what content is requested and from which line-card's faces, this ensures a single outstanding Interest packet in the presence of concurrent content requests as well as it allows to multicast the Data packet received as a response. Thus, an efficient design of the PIT is a key in order to prevent a central bottleneck and enables an NDN at wire speed. However, moving to multiple decentralized PITs rather than a central PIT can be quite challenging to maintain loop detection, correct Interest packet aggregation, and multipath support. Therefore, Input line-card placement lacks loop detection to correct Interest packet aggregation since each PIT is only aware of the local list faces and $list_{nonces}$. In addition, input line-card placement requires a number of lookups in the PIT for the subsequent returning of Data packet which could also result into a bottleneck.

In output line-card placement, only its placement is achieved which requires an FIB lookup per Interest packet even though when the previous Interest packet for the same content has already been received. Finally, loops cannot be detected as each output PIT is only aware of the local $list_{nonces}$. Whereas input/out line-card placement, loops prove difficult to be identified for the same reasons as above. A minor issue is that the Data packet may trigger two or more operations of lookup in the PIT of the line-card where it is obtained, and in the PIT(s) of the line-card(s) from where it was initially subscribed for. Finally, the drawback of the third party placement is that it progenerates the operation through additional switching operations for both the Data and Interest packets.

Table 2.2
PIT Placement Techniques Summaries

Technique	Focus on	Environment	Contributions	Drawback
Input line-card [68, 54]	Placement	Simulation and real test implementation	i) PIT places at each input line-card. ii) Data packet is broadcasted to all input line-cards. iii) Enables multi-path.	i) Lack loop detection. ii) Lack correct Interest aggregation. iii) Requires a number of lookups PITs at each input line-cards.
Output line-card [46]	Placement	Emulation	i) PIT places at each output line-card. ii) Aggregating Interests. iii) Requires a FIB lookup per Interest. iv) Avoiding the problem of cumbersome synchronization among multiple PITs on line-cards.	i) Limitations at case of multi-path. ii) Loops cannot be detected as each output PIT is only aware of the local $list_{nonces}$.
Input/Output line-card [46]	Placement	Emulation	i) PIT places in both input line-card and output line-card. ii) Interest packet makes a PIT entry for both the input line-card at the received and at the output line-card at the sent. iii) Absence of the superfluous look-ups in the FIB. iv) Copied packets in presence of multi-path.	i) Suffers from the latter multi-path issue. ii) Loops prove difficult to be identified. iii) Data may trigger two or more operations of lookup in the PIT of the line-card.
Third Part line-card [55]	Placement	Real hardware	i) PIT places at each input line-card as in the input-only placement. ii) Placement for both multi-path and loop detection. iii) Placement only requires a sole lookup per PIT's operation.	i) It progenerates the operation through additional switching operations for both the Data and Interest.

2.6.3 PIT Replacement Policies

Replacement policy in PIT is one of the important factors that determines the effectiveness of cache (e.g., PIT). It has become even more important with the advent of the technological trends towards highly associative cache practices. The state-of-the-art processors therefore employ various cache policies, indicating that there is no common replacement that stands out as the best [76]. Hence, this section explores some common PIT replacement policies in greater perspective. It is therefore paramount to investigate the performance of different PIT in relations to replacement policies for contemporary workload in different PIT configurations. This will address how some existing policies relate to PIT. In addition, replacement policies have a different effect on the instruction and entries in PIT. This specific problem is addressed by performing a critical evaluation of some possible PIT replacement policies i.e., Persistent [47], Random [47, 76] and LRU [174, 175, 76].

2.6.3.1 Persistent Replacement Policy

Persistent is assumed as the default replacement police for PIT in NDN router which is added to speed up the operations, thus reducing the complexity in implementation [47]. Persistent policy has been proposed as a solution to coordinate PIT in Interest packet interactions. Persistent replacement policy requires that mostly new Interest packet should be selected to be rejected only from its allocated memory space when there is no free space in its PIT. Therefore, this policy does not require holding any information about the access in history.

2.6.3.2 Random Replacement Policy

As the name implies, this policy randomly selects a candidate entry and evict it in order to present free space when necessary. Like the previous policy, this policy does not require holding any information about the access in history. The replacement policy chosen here is the simplest; the evicted Interest packet is randomly selected. This policy seems pretty easy in implementation due to the advantage of a pseudo-random counter for the whole cache operation. It however, consumes few resources, but it performs badly due it is not usage-based. Its performance relies solely on real randomness of the sequence. In Random policy, when PIT reaches its limit, random entry (could be the newly created one) will be removed from the PIT. Therefore, there are two inverse cases in this policy. The best case is by removing the entry which has too long expiration time with a minimum frequency. On the other hand, the worst case is by removing the entry which has the lowest expiration time with a maximum frequency.

2.6.3.3 Least Recently Used Replacement Policy

The Least Recently Used (LRU) [176] policy is among the most popular algorithms that are based on the least-recently-used cache replacement rule. LRU tends to keep more frequent items used in the cache as well as quick adaptation to the potential changes in document popularity. This results in an efficient performance of the overall replacement policy. In order to understand further the insight into network caching designing and algorithms, it is important to gain a thorough comprehension of the baseline LRU cache replacement policy [177]. In the analysis of LRU caching policy, due to being recently in use, the entry with the highest access is most likely to be accessed again in the near future, and the entry that has been "least recently used" would be replaced by the PIT controller when the PIT demands a new entry adding. Hence, this policy does require holding the information about the access in history. The comparison in Table 2.3 presents a summary of the characteristics of the PIT replacement policies.

One of the primary goals of the NDN is to manage cache contents, particularly in

Table 2.3
PIT Replacement Techniques Summaries

Technique	Focus on	Environment	Contributions	Drawback
Persistent [47]	Replacement	Real test implementation in CCNx	i) Simple to implement and easy to operate. ii) Rejected the new entries when the PIT size reached its limit. iii) Speeding up operations thereby reducing the complexity in time. iv) Not require holding any information about the access in history.	i) Memory space in PIT may be well underutilized which is lead relatively poor performance. ii) Not consideration about Interest lifetime, Interest frequency. iii) Increase the packets drop.
Random [47, 76]	Replacement	Real test implementation in CCNx	i) Randomly selects a candidate entry. ii) Not require holding any information about the access in history. iii) Easy in implementation due to the advantage of a pseudo-random counter.	i) Less efficient than the other policy. ii) Not consideration about Interest lifetime, Interest frequency. iii) Entries may accumulate large request counts is replacing. iv) Always evict the newly.
LRU [174, 175, 76]	Replacement	Real test implementation in CCNx	i) Removing the entry that has not been accessed for the longest period of time. ii) Efficient performance in low traffic congestion. iii) Require holding any information about the access in history.	i) Lead to many unnecessary cache replacement. ii) More expensive and harder to implement. iii) Not care about entry lifetime. iv) Negative impact on the average access time. v) Not efficient via heavy load.

PIT, in NDN routers to accommodate Interest packet efficiently. Similarly, it is also a challenging task to decide which content should be evicted from the memory of the NDN router, and the need arises when a new packet arrives and the cache of the router is full [178]. However, there is a weakness that needs to be developed, and deployment aspects call for an in-depth investigation. Although the Persistent policy is simple to implement and easy to operate, it rejects the new entries when the PIT size reaches its limit. However, no consideration was given to the Interest packet lifetime and Interest packet frequency which can affect the performance of PIT as well as the whole network, In addition, Interest packet aggregation will become useless.

Although the Random policy is simple to implement in hardware, it is less efficient than the other policy because some entries may accumulate large request counts. Random policy may also have little Interest packet lifetime as a factor that is replacing and it could be the newly created one to be removed from PIT. In case of LRU and the number of Interest packet to keep track of increases, it becomes more expensive (i.e., harder to implement and slower) if one wants to ensure that the policy always discards the least recently used entry. LRU can also lead to many unnecessary cache replace [112]. In addition, it is not efficient via heavy load, and does not care about entry lifetime as a factor.

Therefore, the needs to develop a new PIT replacement policy have become so crucial for the existence of many content replacement policies. In order that, our proposes policy is considered the entry lifetime and the popularity of the entry as factors to determine the PIT entry to be replaced, since the spent time and number of requests for each entry of the PIT memory can affect the PIT utilization.

2.6.4 Adaptive Interest Lifetime Techniques

The size and shape of PIT are directly dependent on the traffic that is processed by an NDN routers [179]. In NDN, subscribers transmit Interest packets that establish soft states in forwarding nodes. Hence, Data packets can travel in the reverse path back to subscribers. Each Interest packet has a lifetime that determines how long a soft state is kept valid at maximum, i.e., if no Data packet is received in return. Since Interest packets are only forwarded upstream if the same Interest packet is not already forwarded, the Interest packet lifetime has a direct impact when Interest packets are retransmitted at subscribers [69]. In addition, NDN packets processing is based on an FIB to forward Interest packets towards content sources, and PIT, to remember already sent Interest packets for the duration of an Interest packet lifetime or less if satisfied by Data packet earlier.

This timer for new PIT entry is initialized and set according to the lifetime field in Interest packet. The default time of Interest packet lifetime in NDN is setting 4 seconds (4000 ms) [61, 58]. This is thus considered as the sufficient duration for NDN router to hold an entry in its PIT to receive Data packet. However, in the case of high traffic load or when Data packets are returned with a delay, this may lead to maximum number attaining of entries in the PIT. Consequently, it may result in bottlenecks by exhausting memory space as well as the Interest packet search and propagation delay for waiting longer to download a content. The size of PIT is conclusively said to be directly proportional to the entry lifetime. The problem might be exacerbated by a massive usage of long Interest packet lifetimes, which would further increase the number of simultaneous entries in the PIT.

As for this fact, this table may be overflow, which could also lead to dropping incoming new Interest packet and redundant unnecessary retransmissions of Interest packet.

This can consume most of the bandwidth which may even result in consequent service disruption. In order to avoid the resulted bottleneck experienced by PIT size, the Interest packet lifetime should be adaptive. To address the problem, several adaptive lifetime mechanisms have seen proposed, such as DPEL, CCNTimer and LTI. This research carefully reviews each of the above mentioned lifetime mechanisms in wider perspectives.

2.6.4.1 Dynamic PIT Entry Lifetime (DPEL)

Bouk *et al.* in [61] proposed a new scheme named Dynamic PIT Entry Lifetime (DPEL) that dynamically adjusts the Interest packet lifetime in PIT for vehicular NDN. In DPEL, the lifetime of each Interest packet is needed to be calculated at every traversed hop in a distributed fashion in order to acquire the resulting lifetime in an alleviated form if the numbers of relay nodes and hops are increasing. Since the network is always deployed as a highly non-static environment when Interest packets are not satisfied, the data provider and forwarding seem hard to assume that the Data packet will traverse on the same reverse route. Therefore, the resulting profession of the PIT by that Interest packet is termed ineffective. Thus should be deleted earlier than the normal or static lifetime of any Interest packet.

In DPEL, the PIT Entry Lifetime (PEL) of new PIT entry is dynamically adjusted in the PIT of each NDN router that forwards Interest packet. The Interest packet's PEL at a successive forwarder is computed using Interest packet's hop-count and forwarder node's own Interest Satisfaction Rate (ISR) information. The results demonstrate that any forwarding node with a higher ISR has a very small average PEL because it frequently purges PIT entries after receiving the requested Data packets. As a result, a forwarding node with a higher ISR does not need to store PIT entry for a longer time. To support this claim, PEL mechanism was proposed as described below:

The decay rate of PEL of an Interest packet generated for content c, $\mathscr{T}_c$, at hop h is calculate as:

$$\frac{\mathscr{T}_c}{h} = -\mathscr{T}_c \,.\, \mu \,.\, (1 - ISR) \tag{2.1}$$

where μ is the decay constant, $\mathscr{T}_c$ is the PEL at vehicle V and ISR is the Interest satisfaction rate of the Interest packet forwarding node at hop h . The hop number h is incremented and updated by every upstream Interest packet forwarding node. By solving the Equation (2.1), the resulting solution is:

$$\mathscr{T}_c(h) = \mathscr{T}_{c_h} = \mathscr{T}_0 \,.\, e^{-\mu . h . (1 - ISR)} \tag{2.2}$$

where $\mathscr{T}_0$ is the initial decay value assigned by the V_c. At the network setup time, the ISR of all the vehicles is set to 0. In this case, the proposed model also ensures the exponential decay in PEL μ at hop h, as:

$$\mathscr{T}_{c_h} = \mathscr{T}_0 \,.\, e^{-\mu . h} \tag{2.3}$$

The $\mathscr{T}_c$ and μ are initialized by the V_c and PEL depends on these values along with the Interest packet forwarding node's ISR and hop-count h of the Interest packet. Hence, DPEL quickly discards the pending Interest packets received from the subscriber at a significant distance to alleviate the PIT size without compromising ISR.

2.6.4.2 CCNTimer

CCNTimer is an adaptive Interest packet lifetime for wireless multihop communication [69]. CCNTimer algorithm uses the same exponential moving average for $sRTT$

and *rttVAR* as TCP (as referred by Allman *et al.* in [180]). RTT measures the time between the transmission of the first Interest packet in a segment and the reception of the corresponding segment. In case of timeouts, Interest packets may be retransmitted, but the RTT start time of the corresponding segment is not changed because the segment has possibly been successfully transmitted over some hops and cached in NDN routers. T_{out} defines the number of timeouts for each segment. In addition, $sRTT$ is weighted by a factor w to obtain the Interest packet lifetime. This ensures that the Interest packet lifetime is slightly longer than the usual RTT, which avoids spurious retransmissions. While too long Interest packet lifetimes only have an influence in case of collisions, too short Interest packet lifetimes may already trigger retransmissions if Data packets are returned with a delay, thus causing unnecessary traffic. In the worst case scenario, too early Interest packet retransmissions may cause collisions with delayed Data packets. Moreover, possibilities of the collisions of received and transmitted packets are likely to occur regularly in wireless multi-hop networks. Even though the traffic load is low, the Interest packet lifetime is not drastically increased in case of timeouts.

2.6.4.3 Interest Control Protocol (ICP)

Carofiglio, Gallo, and Muscariello in [70], outlined a window-based Interest flow Control Protocol (ICP) driven by an Additive Increase Multiplicative Decrease (AIMD) mechanism to direct and manage the Interest packet rate from the perspectives of the receiver. An ICP is thus imagined as a guide to control Data packet recoveries from the different routes (sub-paths) that constitute every traversing route from a subscriber to a content repository. ICP timer properly sets the Interest packet timer value ($\mathscr{T}$) and the PIT timer, which allow for the removal of the pending Interest packet at nodes to be essential in NDN manageability. Generally, $\mathscr{T}$ must be trivially set larger than the minimum network delay, otherwise every sent Interest packet would trigger a timer expiration. Consequently, a minimum $\mathscr{T}$ is of the essence in order to guarantee higher utilization of the available bandwidth. Hence, in the case of variable $\mathscr{T}$, ICP maintains a delay in round trip to estimate its Data packet receiving which is done by updating RTT_{min} and RTT_{max} averaged over a history excluding retransmitted Interest packets. Table 2.4 summaries the characteristics of all the mechanisms discussed in this sections.

A comparative Table 2.4 presented to complete all idea on the adaptive Interest packet lifetime mechanisms in NDN. DPEL quickly discards the pending Interest packets received from the subscriber at a significant distance to alleviate the PIT size without compromising Interest satisfaction rate. It is implemented for mitigating approximate (40%, 55% and 77%) of PEL and the number entries of PIT for $(\mathscr{T}_c = 0.5, 0.3, and\, 0.1)$, respectively. However, this scheme has given an insufficient explanation regarding the increased number of Interest packet that is used in evaluation. Also, the effect of the PIT overflow on Interest packet lifetime is not considered. Hence, this can affect network utilization. Moreover, the adapting of Interest packet lifetimes is a function under the subscriber responsibility.

CCN Timer performed on two evaluation scenarios (i.e., low traffic and high traffic scenarios). In case of low traffic, CCN Timer outperforms in significantly shorter retrieval times to compare with TCP's RTO and *TimeoutEstimator* without transmitting more Interest packets. In high traffic, CCN Timer performs similar to *TimeoutEstimator* and TCP's RTO according to the transmission times as well as transmitted Interest packets. In addition, the Interest packet lifetime is not required to be doubled in case of a timeout. Evaluations have shown that increasing the Interest packet lifetime only slightly CCNTimer is enough to receive subsequent segments without timeouts. However, this algorithm is applied in simple environment

Table 2.4
PIT Entry Lifetime Techniques Summaries

Technique	Environment	Performance	Contributions	Drawback
DPEL [61]	Simulation	PIT entry lifetime, Avg. No. of PIT , Interest Satisfaction Rate	i) PIT newly entry is dynamically adjusted in the PIT that forwards Interest. ii) With higher ISR does not need to store PIT entry for a longer time. iii) NDN node with higher ISR has a very small average PEL.	i) The effect the PIT overflow on Interest lifetime is not consideration. ii) No. of Interest using in evaluation. iii) Subscriber be responsible for adapting the Interest packet lifetimes.
CCNTimer [69]	Emulation	Content transmission times, retrieval times, Interest satisfaction rate	i) Beneficial in wireless environments. ii) Control quickly Interest retransmissions. iii) Using moving average for rttVAR and sRTT as TCP. iv) Constant Interest lifetime of 4 seconds. v) Not required to double the Interest lifetime in case of a timeout	i) Subscriber be responsible for adapting the Interest packet lifetimes. ii) The effect the PIT overflow on Interest lifetime is not consideration. iii) Apply the algorithms in simple environment settings.
ICP [70]	Analytically	Delivery time, queue delay, and average time interval	i) Realizing a window-based Interest flow control. ii) Analyzing ICP on a multi-bottleneck and single link scenario. iii) Outstanding Interests: Interest Window Increase and Interest Window Decrease.	i) Not evaluate the No. of unnecessary retransmissions. ii) Subscriber be responsible for adapting the Interest lifetimes. iii) Effecting PIT overflow on Interest lifetime does not consideration.

settings. Thus, the subscriber is responsible for adapting the Interest packet lifetimes as a function. Moreover, the effect of the PIT overflow on Interest packet lifetime on the performance of PIT was not mentioned.

ICP analysis is realized as the most advantageous statistical bandwidth sharing and for guaranteeing efficient and fairer bandwidth utilization. However, this protocol does not evaluate the number of unnecessary retransmissions as a metrics in order to evaluate this protocol since this parameter is very important for determining the effect of adapting Interest packet lifetime on Interest packet flow control. Also, the affecting PIT overflow on Interest packet lifetime was not considered. In addition, the subscriber was responsible to adapt the lifetimes of Interest packet when the congestion happens.

Therefore, the introduction of smarter mechanism for an Interest packet lifetime management at content routers, which adapt Interest packet's lifetime as a function of the network load: a router can grant larger lifetime value for incoming Interest packet in case of low traffic, and PIT is not overflowed, and conversely, implement a short of lifetime value for incoming Interest packet shaping when the traffic is increased and PIT is overflowed. With such a mechanism, our mechanism will break down the hypothesis that the NDN routers do not manage Interest packet lifetime values.

2.7 Theories Pertinent to PIT Management

There are three major theories adopted in this research; Queuing Theory, Renewal Theory and Scheduling Theory. The following subsections will introduce these theories in the context of this research

2.7.1 Queuing Theory

Queuing theory is the key analytical modeling technique used for computer systems performance analysis [181]. Queuing theory deals with stochastic models that depict the transformation of the subscribers' random flows during servicing by servers. In such a way, Queuing theory is significant whatever the concept of queue, breakdown, waiting, and loss that appear [182]. At this point, a Queue Management algorithm [183] is the process where a router chooses when to drop a packet and which packet should be selected for dropping at its output port, when it becomes congested. Queue Management algorithms attempt to approximate fairness by appropriately dropping packets in order to minimize network congestion and keep up reasonable queue lengths. One case of queue management algorithm is the Active Queue Management (AQM) (as discussed by Thiruchelvi and Raja in [184]; as discussed elsewhere [183, 184, 185, 186]), which tries to balance congestion control at the endpoints to avoid packet dropping.

Characteristically, AQM is a proactive congestion control mechanism, where the data are sent by a network node to the sources if early congestion is detected. The data can be sent expressly as "Explicit Congestion Notification (ECN)" (as discussed in [187]) marks, or verifiable by packet drops [188]. When congestion increases, the AQM scheme intensifies its feedback to the Transport Control Protocol (TCP) endpoints, i.e., by dropping or marking more packets. The sources, in response to these congestion notifications, would decrease their data transmission rates in order to avoid queuing overflows and reduce the losses that can result. Subsequently, it is necessary that the AQM must readily detect congestion, and give quick and compelling feedback to the sources. Generally, the queue management mechanisms consist of three mechanisms, namely:

- The congestion indicator, and
- The congestion control function, and
- The feedback method.

The *congestion indicator* is utilized by the queue management mechanism to decide if there is congestion, whereas the *congestion control* chooses what needs to be done if congestion is identified. The *feedback method* is the congestion flag which is used for alerting the source to adjust its transmission rates [183]. According to Thiruchelvi and Raja in [184], AQM can be classified into three families, namely queue length based (e.g., Random Early Detection (RED) [189], Stabilized Random Early Drop (SRED) [190]), rate based (e.g., Adaptive Virtual Queue (AVQ) [191], Stabilized Adaptive Virtual Queue (SAVQ) [192]) and AQM based on queue length and rate merits (e.g., Random Exponential Marking (REM) [193], Stabilized Virtual Buffer (SVB) [194]).

Adaptive Virtual Queue (AVQ) is a rate-based AQM scheme [195, 191, 196, 197], which attempts to maintain the arrival rate at a desired usage. According to Adams *et al.* in [183], the AVQ is based on the penalty function technique as proposed by Kunniyur *et al.* in [191]. The AQM-AVQ can be adapted to address the PIT overflow problem by proposing the Adaptive Virtual Pending Interest Table (AVPIT) mechanism.

2.7.2 Renewal Theory

The Renewal theory is primarily concerned with the Poisson processes for arbitrary holding time (as discussed by Miaji in [198] and by Couch in [199]). This definition widens the range of utilizing the Renewal theory in packet scheduling mechanism more than the Probability theory. In other words, Renewal theory participates in calculating the expected time for a packet to arrive in a specific queue and compares the long tenure benefit of different insurance policies. Another distinguished property of Renewal theory is the positive and negative rewards, which imply that there are some benefits or a better allocation for the good behaved user and some charges or drawbacks for the misbehaved users [198]. The Renewal process is considered as a continuous process as long as the arriving process is independent or uncorrelated. The concepts of Renewal Theory can be adapted to address the impact of Interest packet lifetime on the performance of PIT by proposing the Smart Threshold Interest Lifetime (STIL) mechanism.

2.7.3 Scheduling Theory

In computer networks, scheduling is the method by which work specified by some means is assigned to resources that complete the task (job). The task may be virtual computation elements (e.g., share CPU time, threads, and data flows), which are in turn scheduled onto hardware resources (e.g., network links, processors, and expansion cards). The main purpose of scheduling theory is to minimize resource starvation and to ensure fairness amongst the parties utilizing these resources. Scheduling deals with the problem of deciding which of the outstanding requests is to be allocated resources. There are many scheduling algorithms, such as First In First Out, Custom Queuing, Weighted Fair Queuing, and Priority Queuing [198].

Priority Queuing (PQ) is a congestion management technique that schedules traffic, for example the higher-priority queues "always" get serviced first. This can cause the traffic of other lower-priority queues to starve out. The basic idea of PQ is to isolate the traffic in various classes concurring to a particular rule [200]. The theory of PQ suggests that (t) the total amount of processing required by the entry waiting to be processed (or, in queuing terminology, the clients holding up in queue) has already been determined [201]. The Highest Lifetime Least Request (HLLR) policy uses the concept of PQ to determine the PIT entry that needs to be replaced for efficient PIT utilization.

2.8 Overview of Applied Evaluation Methodology

In this section, various techniques are presented and considerations for evaluating different ICN designs. Through this, practitioners and researchers alike would be able to contrast and compare different ICN architectures each other. In addition, introduced several topologies that have been used in ICN studies so far, which are adopted and selected them in this study.

2.8.1 ICN Simulators Tools

Simulation has been widely the option of simulating dynamic scenarios, mainly networks and real systems. It is a computer-based system model or generated using computer programming. In addition, simulation is a more flexible tool for studying the performance of various protocols [202]. Thus, simulation was the chosen method for the performance evaluation in this study as this technique is widely used in representing the dynamic behavior and responses of real systems [203]. Many discrete event network simulators are available for use today. Some of these are commercial products requiring researchers to purchase them first, whereas others are free and open source products that can be downloaded, used, and modified. Some of the most popular tools that are available for communication network researchers are OPNET [204], OMNET++ [205, 206], and Network Simulator 3 [207, 208].

- **OPNET**: The Optimized Network Engineering Tool (OPNET) (as discussed by Chang [204]; and as discussed elsewhere [209]) is a very powerful network simulator. Its main purposes are to optimize cost, performance, and availability. OPNET provides an environment that supports modelling of communication networks and distributed systems. The OPNET environment contains tools for all phases of a study, including design, simulation, data analysis, and data collection. There are three layers for the hierarchical structure of an OPNET model: Network layer, Node layer, and Process layer. Each of these layers has an editor incorporated with them in the OPNET environment. According to Chang in [204], the major features of OPNET are explained here as: *(i) Modeling and Simulation Cycle*, where OPNET gives effective tools to help clients experience three out of the five stages in a design circle (i.e., the working of models, the execution of a reproduction and the investigation of the output information). *(ii) Hierarchical Modeling*, OPNET utilizes a various leveled structure to modeling. Each level of the hierarchy depicts distinctive parts of the complete model being recreated. *(iii) Specialized in communication networks*, detailed library models give backing to existing conventions and permit scientists and engineers to either adjust these current models or grow new models of their own. *(iv) Automatic simulation generation*, OPNET models can be gathered into executable code. An executable discrete-event simulation might be fixed or simply executed, bringing about the output information.

- **OMNeT++:** OMNeT++ [210] is a C++ based discrete-event simulator for modeling correspondence systems, multiprocessors and other disseminated or parallel frameworks. It is an open source, and can be utilized under the Academic Public License that makes the product free for non-benefit utilizing. The inspiration of creating OMNeT++ was to deliver an effective open-source discrete-event simulation tool that can be utilized by scholastic, educational and research-oriented commercial institutions for the simulation of computer networks and distributed or parallel frameworks. OMNeT++ is accessible on every common platform, including Linux, Mac OS/X and Windows, using the GNU C Compiler (GCC) tool chain or the Microsoft Visual C++ compiler. OMNeT++ represents a framework approach. Instead of specifically giving reproduction parts to computer networks, queuing networks or different areas, it gives the fundamental apparatus and instruments to compose such simulations. Particular application ranges are upheld by different simulation models and frameworks, such as the Mobility Framework or the Internet NETtworking (INET) Framework. These models are created totally freely of OMNeT++ and take after their own release cycles. An OMNeT++ model comprises of modules that communicate with message passing. The dynamic modules are named straightforward modules; they are composed in C++ utilizing the simulation class library. Basic modules can be gathered into compound modules and so forth; the number of hierarchy levels is not constrained [205].

- **NS3:** Network Simulation 3 (NS3) is a discrete event simulator, which can be used for the implementation of numerous applications. The NS3 project started in [211] as an open source. It is a free and open source network simulator that has been made available for teaching, research community, students, and development work under the General Public License, version 2 (GPLv2) license [212]. In NS3, there are numerous external animators and tools. This simulation platform provides users with a single, integrated Graphical User Interface (GUI) environment, data analysis, and visualization [213]. NS3 has been designed in modular fashion as a set of libraries. These libraries may be combined together as well as with other external software libraries. Currently, NS3 can be installed only on Linux operating system in the native mode. In addition, the non-availability of backward compatibility with NS2 will also hinder the ready acceptance of NS3 as the default simulation tool since users will be reluctant to abandon NS2 immediately [203].

According to Pentikousis *et al.* in [214] and Pentikousis *et al.* in [215], respectively, simulators and emulators must be able to capture faithfully all features and operations of the respective information-centric architectures. NDN architecture needs flexible simulators that would support user's easy usability, configurability, simplicity, and logical programming dynamism to simulate NDN. NDN is required that simulators are easily extendable through open source options. Adequate documentation and manual guides are needed to easily handle simple and complex network experiments through increasing network sizes, nodes, time, parameters, and metrics selection. There are some simulators for CCN/NDN architecture:

- **CCNPL-Sim,** this simulator is written in C++ [216] and is based on CBCBsim, from which it imports part of the forwarding layer and the Combined Broadcast and Content-Based (CBCB) routing protocol, whereas the features of CCN protocol have been designed from the scratch. The simulator has been considered to evaluate per-hop sending behavior and publisher based congestion control, where a fine-grained control over individual packets is basic to get precise execution results. CCNPL-Sim is the main CCN/NDN simulator to offer out of the box implementations of flow control algorithms, like Additive Increase Multiplicative Decrease (AIMD), thus representing a characteristic decision to maintain a strategic distance from the weight of a comparable usage from the scratch. CCNPL-Sim has the drawback of utilizing a custom discrete-event simulator. Thus, aside from the few scale studies that concentrated on congestion control, different simulators might be desirable over CCNPL-Sim for more extensive purposes or vast scale research for which ccnSim is a superior fit [82].
- **ccnSim** [217] is a kind of simulator which is known as chunk-level simulator for the CCN architecture and is designed in C++ under the framework of Omnet++, that is depicted and traditionalized in. The principle point of ccnSim is versatility, permitting to address situations with expansive CCN cache size (up to 106

chunks) and catalog sizes (up to 108 contents) on off-the-shelf item equipment [82]. According to Afanasyev in [121], ccnSim was composed and executed with the primary objective of executing experimentation of various cache replacement policies for the cache of the NDN router. In such a case, it cannot be considered a fully showcased execution of the current NDN design. In the present adaptation of ccnSim, the FIB and the PIT segments are actualized in the least complex conceivable way. This implies that it is incapable of assessing the diverse information sending procedures, distinctive directing arrangements, or diverse congestion control techniques.

- **ndnSIM** module is part of the NDN tool set, which permits the execution of NS3 simulations (as discussed in [218, 47]). In particular, five versions of ndnSIM are available including ndnSIM 1.0, ndnSIM 2.0 ndnSIM 2.1, ndnSIM 2.2 and ndnSIM 2.3. ndnSIM is based on NS3 structure. The development of ndnSIM has the objectives mentioned below [47]:

  – It is an open-source bundle to empower the exploration group to execute the activities on a typical simulation framework.

  – It is ready to loyally simulate the entire operations that are necessary for the NDN protocol.

  – It maintains packet-level compatibility with CCNx execution [219], to permit sharing of the activity estimation and packet examination tools between CCNx and ndnSIM. In addition. It coordinates utilization of real CCNx traffic traces to drive ndnSIM simulation tests.

  – Being ready to bolster expansive scale simulation tests.

  – Facilitating network layer experimentation with directing, content caching, packet sending, and congestion administration.

Following the NDN design, ndnSIM is actualized as another model of the network layer protocol that may keep executing on top of any accessible model related to available link-layer protocol (e.g., wireless, CSMA, and point to point) as well as on top of a network layer (e.g., 1Pv4 and 1Pv6) and transport layer (e.g., TCP and UDP) protocols. This adaptability permits ndnSIM to reproduce situations of different homogeneous and heterogeneous sending scenarios (e.g., NDN-only and NDN-over-IP).

The simulator is actualized in a particular mold, utilizing separate C++ classes to model behavior of every network layer element in NDN: PIT, FIB, CS, system and application interfaces, Interest forwarding methodologies, and so on. This modular structure permits any segment to be effectively adjusted or supplanted with no or insignificant effect on different parts. In spite of the center protocol stack, ndnSIM incorporates various fundamental activity generator applications and aids classes to rearrange formation of simulation scenarios (e.g., helper to install NDN stack and applications on nodes) and tools to accumulate simulation insights for estimation purposes.

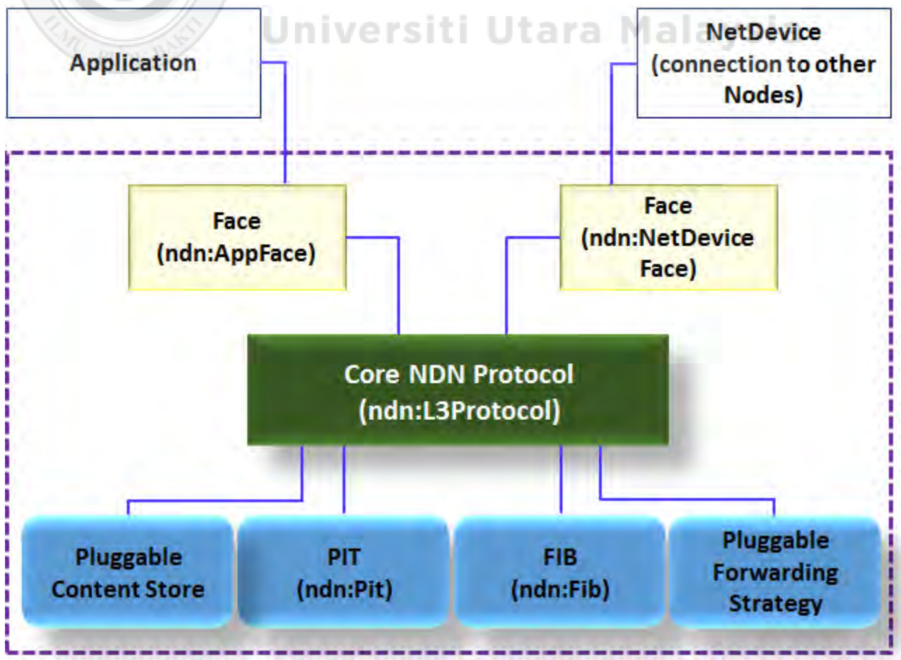


Figure 2.12. ndnSIM Components Block Diagram [47]

The "ndn:L3Protocol, ndn:Face, ndn:ContentStore, ndn:Pit, ndn:Fib, ndn:ForwardingStrategy and the reference to applications related to NDN, containing easy packet generators and sources are component-level constructs which have been deployed in ndnSIM. Figure 2.12 demonstrates the essential collaborations among them. Each element in line with the core exception “ndn::L3Protocol” has a great amount of substitute implementations which might be randomly selected by the simulation scenario with the help of classes, known as “helper classes” (for more details see http://ndnsim.net/helpers.html).

Many simulators are obtainable for CCN/NDN with a corresponding scheme. According to Tortelli *et al.* in [82], Figure 2.13 reports the results of the study, where the label *Custom* stands for both articles in which the authors claim to utilize their custom simulations, and papers. The generic tools, such as NS3, Matlab, Omnet++, and QualNet are mentioned with no sign about the required alterations and/or with no reference to the utilized code. Based on figure below, it is obvious that there is several arrangement of ICN simulation tools, as discussed in the surveyed papers.

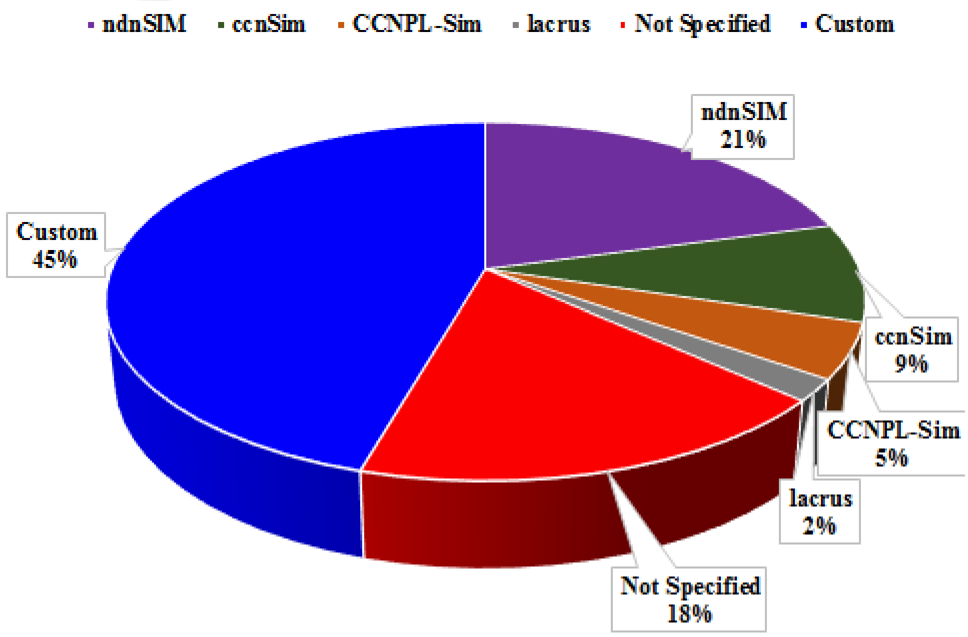


Figure 2.13. ICN Simulation tools used [82]

The dominant part of which is CCN/NDN simulation, the most well-known simulator is ndnSIM with 21%. On the other hand, about 2/3 of presented results are not reproducible, due either the researchers have not even specified the tool used for the evaluation part of their proposal 18%, or they have used a custom simulator 45%. Table 2.5 describes a comparison of the following three simulators: CCNPLSim, ccnSim and ndnSim.

Table 2.5
Comparison Between Different Simulations

Criteria	**CCNPL-Sim**	**ccnSim**	**ndnSIM**
Real code execution	✗	✗	✗
Debugger support	✔	✔	✔
Tracing support	✔	✔	✔
Open source	✔	✔	✔
Based on environment	NS3	OMNeT++	NS3
Scalability	✗	High	High
Deployment	Moderate	Moderate	Moderate

2.8.2 Topology Selection

According to [215], "*there is no single topology that can be used to easily evaluate all aspects of the ICN paradigm*". Studies from [220, 221, 222] have observed that the ideal structure of the topologies can affect the overall performance of the network. This is seen as a result of the overall structure or the hierarchical relationships that exist in the network topologies. In this research, several network topologies with different network sizes and varying number of nodes were used to test the validate and evaluate our proposal models. More specifically, the scenarios are apply an Dumbbell topology [223, 5], Tree topology [223], an Abilene topology [224] and Rocketfuel-mapped AT&T topology [121]. The experiment outcome for every simulation scenario compares the related work simulation results with the model simulation results.

Dumbbell topology is a very common topology that has been used in many congestion network simulations [225]. Several researches and studies [226, 59, 126, 203] have used Dumbbell topology for evaluating the fairness properties of congestion control scheme in CCN/NDN. The former Dumbbell topology was used to study the impact of competing Interest packet rate on links and overflows of the PIT between consumers and many publishers. A simple Dumbbell topology (see Figure 2.14) contains six nodes (i.e., two subscriber nodes, two publisher nodes, and two NDN routers) and five links, is selected [227].

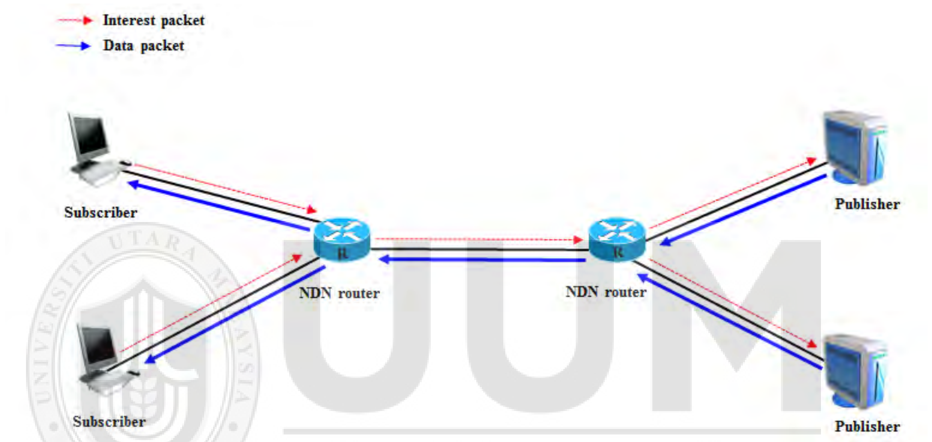


Figure 2.14. Dumbbell Topology [227]

Continuously with Figure 2.15 that shows a simple a Tree topology. A tree topology is a special type of structure in which many connected elements are arranged like the branches of a tree. For example, tree topologies are frequently used to organize the devices in a corporate network. In a tree topology, there can be only one connection between any two connected nodes. Because any two nodes can have only one mutual connection, tree topologies form a natural parent-child hierarchy. Studies by [226, 224, 228, 169] have evaluated them works using Tree topology in order to allow for the expansion of an existing network and achieve deep knowledge for the purpose of this study. In this research, Tree topology contains 25 nodes (i.e., nine subscribers, nine publishers and seven NDN nodes and 24 links) [223].

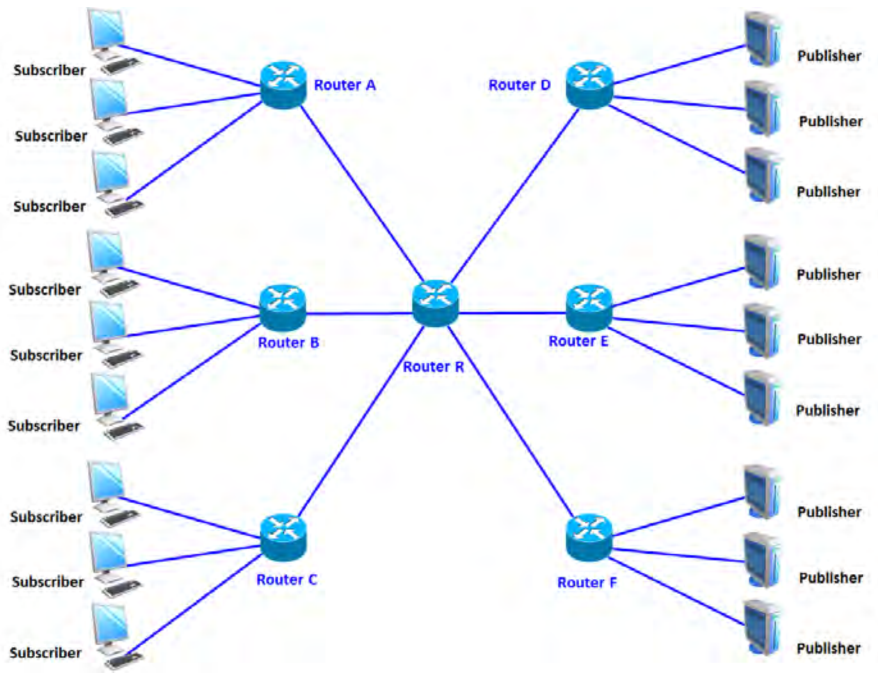


Figure 2.15. Tree Topology [223]

The third topology is based on the Abilene network, which was created by the Internet2 community and connects regional network aggregation points to provide advanced network capabilities to over 230 Internet2 university, corporate, and affiliate member

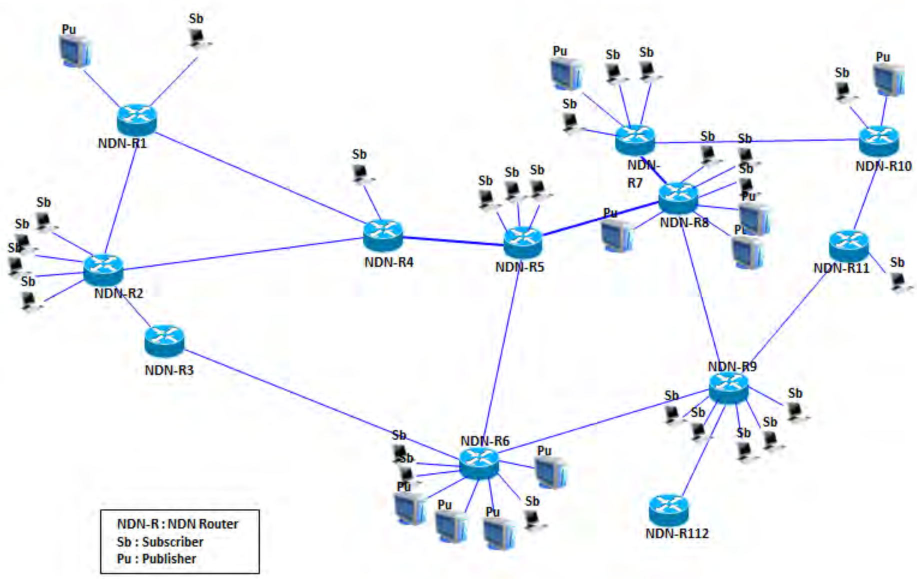


Figure 2.16. Abilene Topology [224]

institutions in the US [229]. Recent studies by [230, 78, 231, 167] have hence submitted the importance of using Abilene topology with more emphasis on the probing result introduced. Abilene topology in this study (see Figure 2.16) consists of 11 NDN routers, 25 consumer nodes, 10 publisher nodes, and 50 links [224].

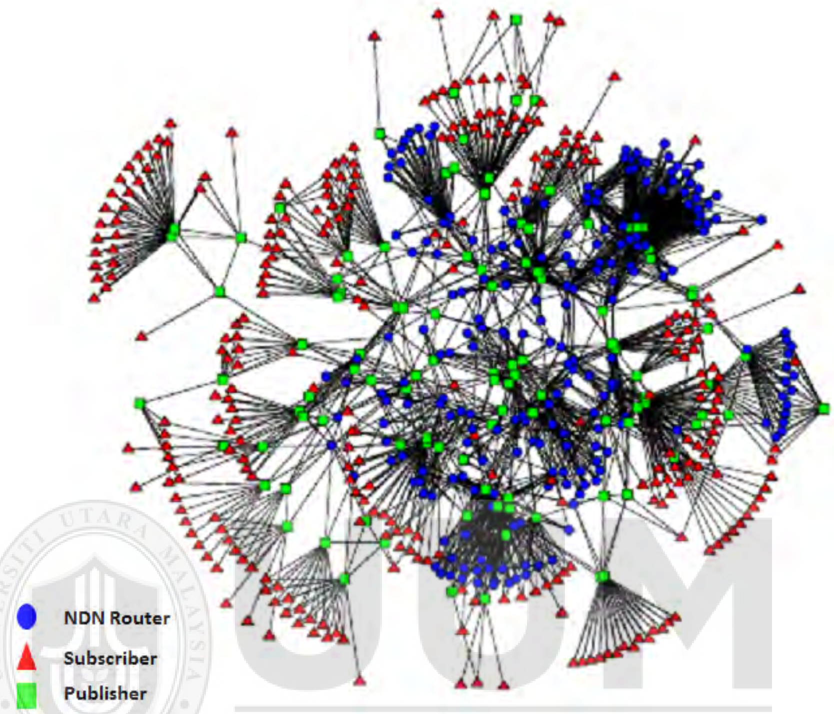


Figure 2.17. Rocketfuel-mapped AT&T Topology [121]

Rocketfuel is very good at collapsing the interfaces on the same router (alias resolution), leading to more accurate maps. It also uses many (over 600 in our data set) public traceroute servers as vantage points, thus providing a fair view of an ISP's routing [232]. Therefore, several studies such as [233, 132, 105, 234, 121, 119, 235] were utilized a Rocketfuel-mapped AT&T topology in their experiments. In order to perform our evaluation model in a more realistic network environment, a modified version of Rocketfuel-mapped AT&T topology was chosen (see Figure 2.17). Rocketfuel-mapped topology nodes is separated into three types: subscriber (296) nodes, NDN routers (108), and publisher (221) nodes and nodes and 2,101 links. Table 2.6 is illustrated the topologies that chosen in the present simulation scenarios based on validation and evaluation the design models.

Table 2.6
Validation and Evaluation the Design Models vs Topologies

Topologies	AVPIT		STIL		HLLR		PITCM	
	Va.	Ev.	Va.	Ev.	Va.	Ev.	Va.	Ev.
Dumbbell Topology	✔							
Tree Topology		✔	✔		✔			
Abilene Topology				✔		✔	✔	✔
Rocketfuel Topology							✔	✔

Va.: validation and Ev.: evaluation

2.9 Summary

This chapter provides the new theoretical shift of Named Data Network (NDN) by covering in details many PIT management techniques. This chapter started with an overview of NDN; the main concept and challenges have been demonstrated in this area. Then, the main issue of PIT that can effect network performance was highlighted. In addition, many techniques in this area were explained in detail based on the literature review techniques since they are related to this research problem and scope. Moreover, a diagnostic evaluation to all related techniques was achieved in order to highlight and identify the gap. As a result, the proposed techniques aimed at mitigating the effect of increasing the Interest packet rate on the PIT and enhancing the utilization of PIT. In the next chapter, the research framework for attaining the objectives of this research, which were highlighted in Chapter One, was presented. The following chapter also involves validating and evaluating PITCM approach.

CHAPTER THREE
RESEARCH METHODOLOGY

The main goal of this research is to manage of Pending Interest Table (PIT) entries and to control the incoming Interest packets in Named Data Network (NDN) architecture for enhancing the utilization of PIT. To realize this goal, Pending Interest Table Control Management (PITCM) approach was proposed, which consists of an Adaptive Virtual Pending Interest Table (AVPIT) mechanism, Smart Threshold Interest (STIL) mechanism, and Highest Lifetime Least Request (HLLR) policy. The combination of the AVPIT, STIL, and HLLR is distinctive of this research.

Hence, this chapter introduces the methodology that was employed in this research to design and implement PITCM, verification and validation of the implementation as well as the performance evaluation process. The chapter starts with the overall research methodology framework as showed in the next section. Section 3.2 shows the initial plans of the research. Section 3.3 describes the second stage that focused on studying and criticizing the previous works. Modeling design is demonstrated in Section 3.4. Section 3.5 presents the experimental design. Verification and validation methods are described in Section 3.6. Section 3.7 proposes the evaluation and analysis of the result, and Section 3.8 presents research conclusion followed by the report documentation in Section 3.9. Finally, in Section 3 .10 the chapter summary is presented.

3.1 Design Research Methodology Framework

This thesis aims at developing management approach for PIT based on queuing, scheduling, and renewal principle that can adapt itself to the changes in the behavior of service PIT. By adapting the changes in the behavior of service PIT, PITCM was designed and implemented which gives results in high utilization of PIT in terms of low-Interest drop, low-delay, low-retransmission, low-PIT length and high- Interest satisfaction that may meet the requires of the subscribers. These requirements are fitted with the design research definition as proposed by Lucienne and Amaresh in [294]. They stated that the research design must be scientific in acquiring valid results in both the theoretical and practical sense. Due to the unique features of the research design, a special methodology is needed.

The research methodology can be either traditional such as Design Research Methodology (DRM) [294], Waterfall methodology [295], spiral methodology [296], Prototype methodology [297], V-Shaped methodology [298], or specific (see [198], [299], [300] and [202]). Thus, the goal of this research is to determine the answers to the questions through the usage of scientific techniques. Therefore, our research methodology that was applied in this work would be scientific and comprehensive enough to guide the entire process from the beginning to the end. This probably ensures that the experiments conducted and the results produced are trustworthy, repeatable and comparable.

The specific research methodology of this thesis consists of several stages. These included an initial plan of this research, a study of the previous works as well as a critical review of them, a design of the model, an experiment of the system design, verification and validation, an evaluation and analysis, research contributions, and reporting the work. Figure 3.1 illustrates the research methodology stages and the link between them with the main process and outcomes for each stage. In the next sections, an explanation of the main concepts and objectives of each stage were provided. In addition, the main methods and deliverable for each stage was emphasized.

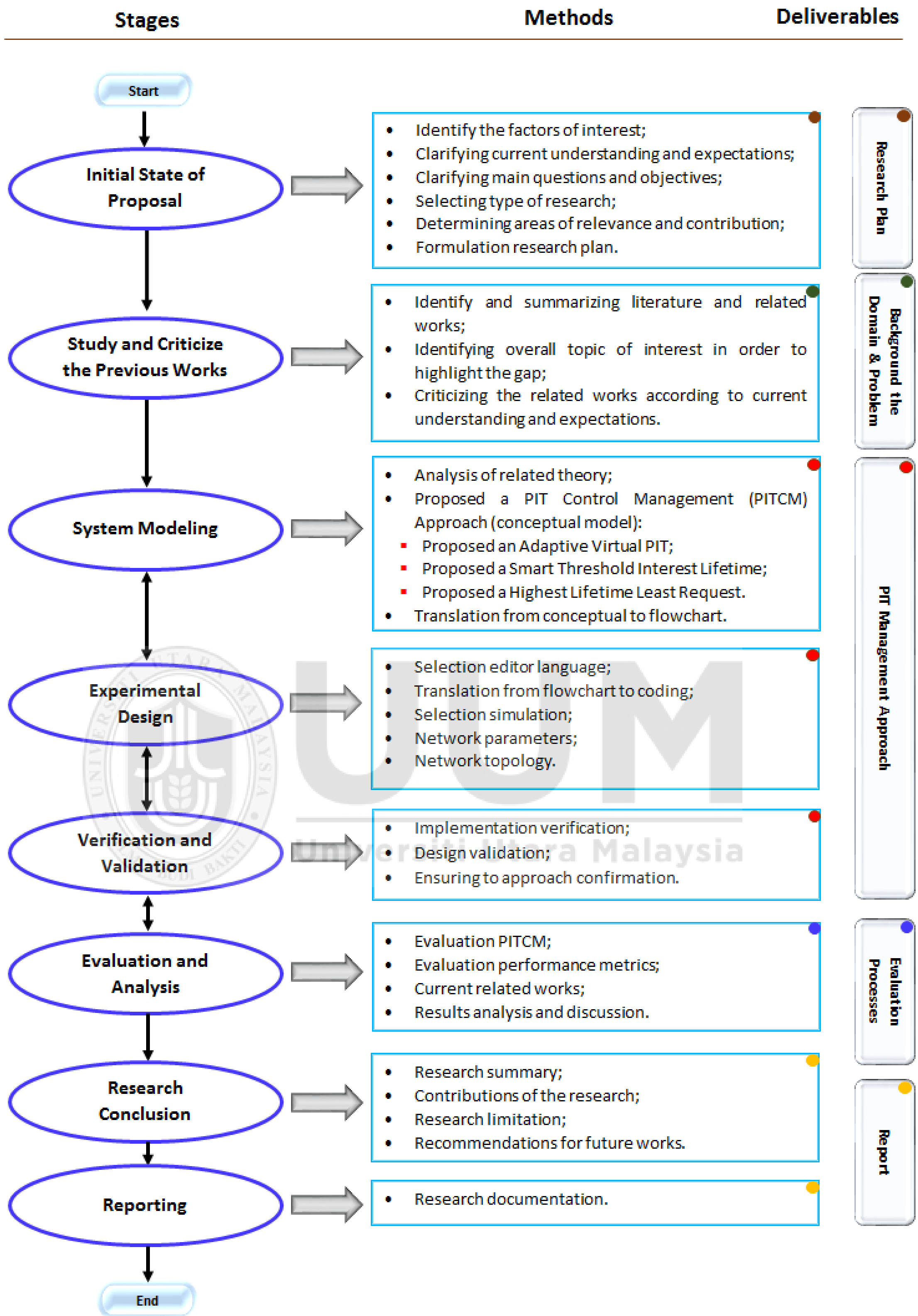


Figure 3.1. Research Methodology Framework

3.2 Initial Plan of Research

The key goal of the research is to present that the solutions propose for a problem are meaningful enough to justify the study. The other goal is to demonstrate that the methodology used is feasible and appropriate, and the outcomes are likely to verify the efficiency and provide an original contribution. In other words, what you are answering is "*will it work?*". Therefore, the first stage in any research is to draw the research plan in order to highlight the main content for this work and how it was distributed. The initial plan of research consists of three phases as depicted in Figure 3.2.

Step One (research focus) studied the literature review and highlighted the challenges of NDN paradigm, the main issues associated with previous works regarding PIT in NDN architecture in order to identify the research gap, the objectives, the scope and the research significance. Step two (core thesis contributions) focused on proposing a Pending Interest Table Control Management (PITCM) approach that has been able to mitigate the overflowing of PIT and manage the PIT entries as well . This approach can support the management of PIT in NDN architecture as well as monitor and control the incoming Interest packet during increasing the number of PIT entries with highest Interest packet lifetime that may cause PIT overflow. In this step, the AVPIT, STIL, and HLLR were designed and used to achieve the purpose of this research.

Integrating AVPIT, STIL and HLLR into PITCM, testing and evaluation the proposed approach is shown in the Third step. This step which includes the performance metrics, validation and network environment, as well as a comparison with the current solutions. In addition, to determine the parameter that should be applied, such as PIT size, the number of Interest rate in the network depends on the number of user's nodes and topologies.

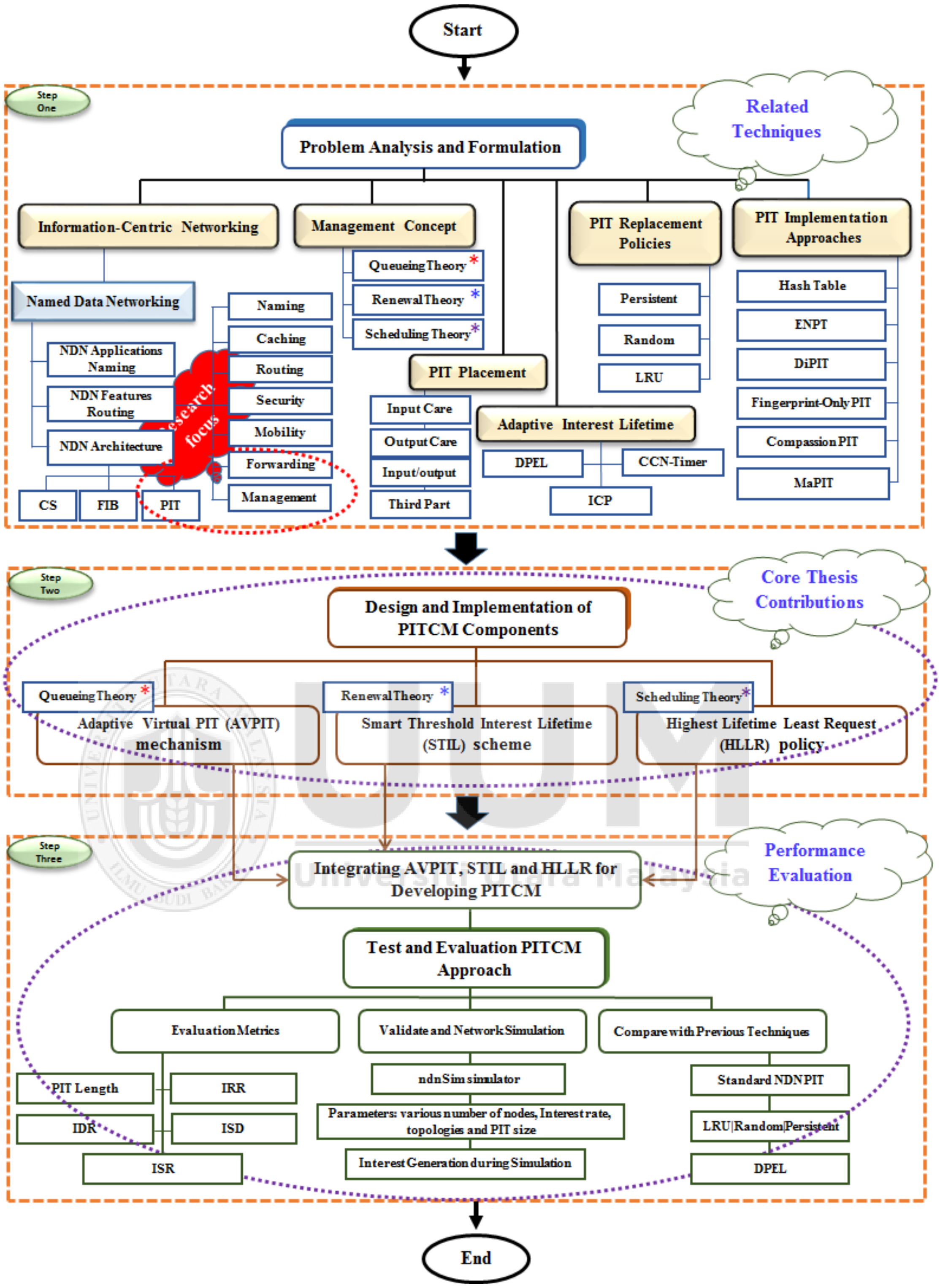


Figure 3.2. Research Plan

The outcome of Initial Plan of Research stage are:

- Forming a research focus and the research motivation, and
- Forming the research problem, the research questions in line with the research objectives, and
- Identifying the research scope, type of research, research methods, and
- Last but not least state out the expected research contribution and deliverables.

3.3 Overview and Critical Analysis

It is the second stage of the research framework, which was used to describe and study the NDN architecture in the literature. In addition, this stage was used to obtain a deep understanding of the PIT management, Interest lifetime, and replacement techniques in order to introduce the gaps that cause the problem of PIT in NDN router for the current work. Hence, the effect of the management of PIT in the performance of entire network is identified and highlighted. Moreover, this stage also criticized the current works that were used to address the problem of PIT. In this stage, the researcher is identified the important key factors discussed the research problem, e.g., the PIT size and the number of entries that let PIT becomes overflow. Figure 3.3 illustrates the main processes in this stage, where each process was provided to increase the understanding that can lead to refining a conceptual framework of this research.

The outcomes of critical analysis stage are:

- Identifying the problem of PIT, and
- Highlighting the management issue in PIT, and
- Criticizing the literature review related to the current PIT management solution (Chapter Two).

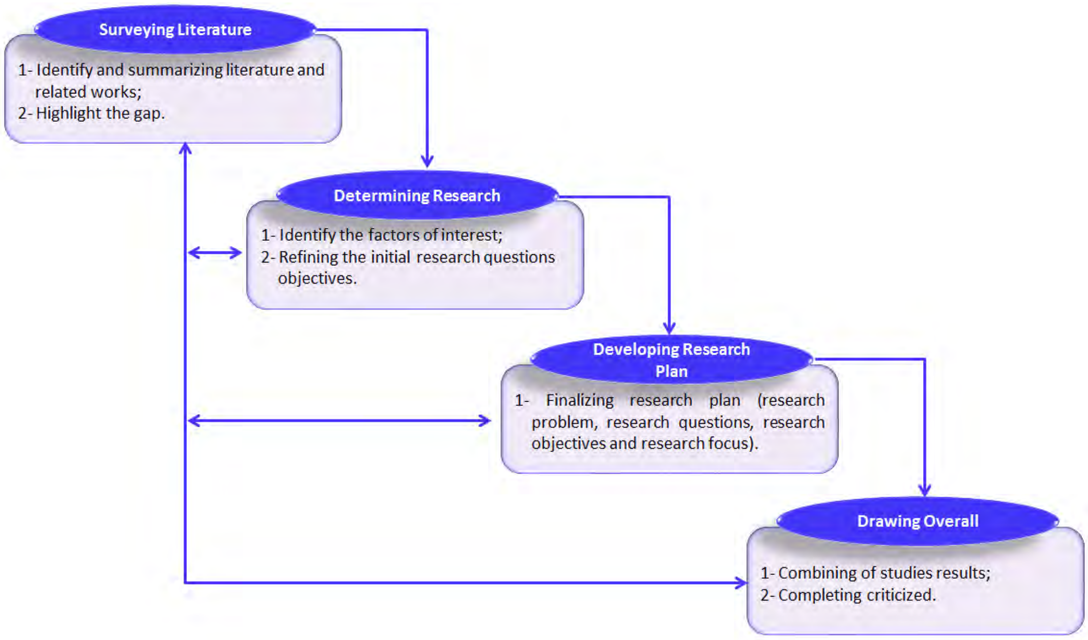


Figure 3.3. Main Steps in the Critical Analysis Stage

3.4 System Modeling

System Modeling considers the main stage in this framework because it includes the proposed approach. The adopted theories are queuing theory, renewal theory, and scheduling theory. These theories are behind for designing our approach components in order to manage the entries in the PIT when it is filled. For example, the queuing theory manages the flow Interest packet. The renewal theory manages the Interest packet lifetime of the incoming Interest packet. Whereas the scheduling theory manages the entries inside the PIT when it became overflow. Thus, Pending Interest Table Control Management (PITCM) approach design first comes from the concept, the problem, the objectives and the scope.

The concept in this research is to propose three major properties, namely Adaptive Virtual Pending Interest Table (AVPIT) mechanism, Smart Threshold Interest Lifetime (STIL) mechanism, and Highest Lifetime Least Request (HLLR) policy of PIT system in order to monitor, manage and control a fast, space-efficient of PIT in NDN router. Thus, these research goals were achieved, as presented in Chapter One. PITCM approach is shown in Figure 3.4.

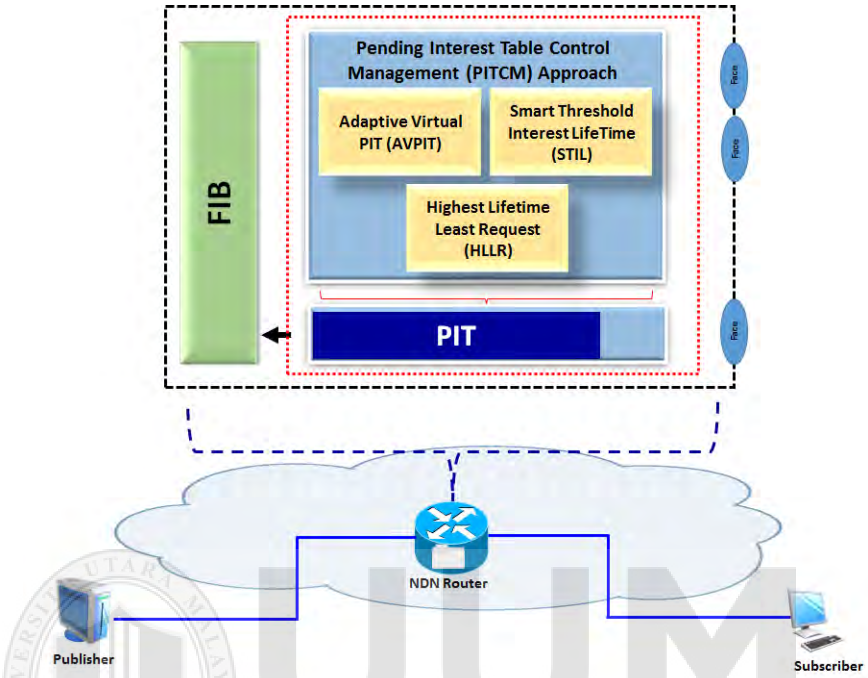


Figure 3.4. PITCM Approach

Queuing theory, renewal theory and scheduling theory build the infrastructure concept for the proposed approach. Therefore, the conceptual design is further analytically explained by the system. Hence, further clarification of the proposed concept is required.

3.4.1 Pending Interest Table Control Management Conceptual Model

In order to improve PIT while basing on the critical review of several related works, a PIT management has been determined as the key factor to be addressed. In addition, the aim of this research were to design and implement a PITCM that manages the PIT to mitigate PIT overflow and increase the utilization of PIT in Named Data Network architecture in order to prevent network collapse. Hence, in addressing the management of PIT, a conceptual framework was designed to clarify the expected improved situation by proposing an AVPIT mechanism, a STIL mechanism, and a HLLR policy. Figure 3.5 illustrates the proposed conceptual model of this research.

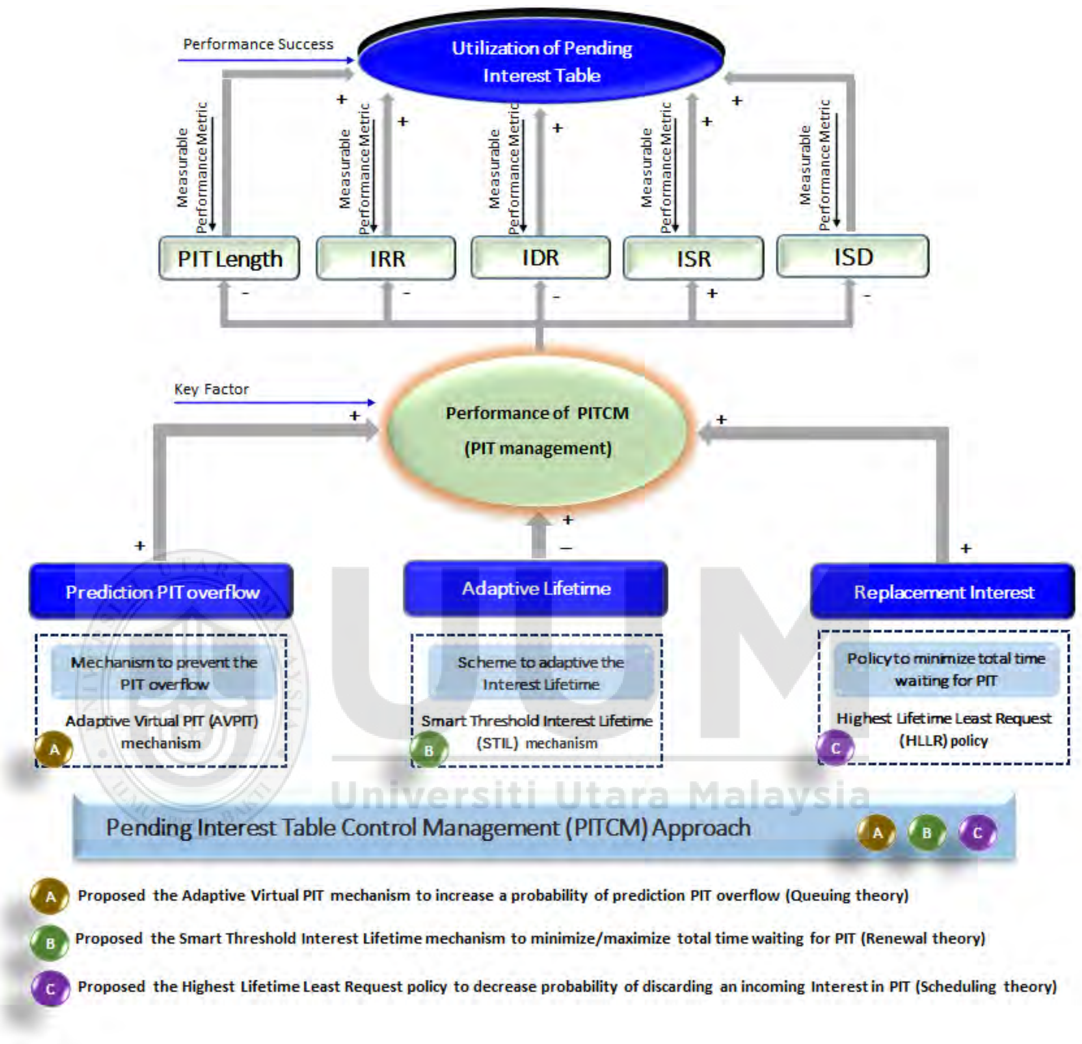


Figure 3.5. Conceptual Model of PITCM

3.4.2 Adaptive Virtual Pending Interest Table Mechanism

Adaptive Virtual Pending Interest Table (AVPIT) mechanism was designed to predict the impending of PIT overflow when the number of PIT entries increases due to the heavy Interest packets load received to the NDN router. An AVPIT uses both PIT size and Interest packet arrival rate as part of its overflow indicator and attempts to keep the Interest arrival rate and the PIT size around their individual target values. Figure

3.6 and Figure 3.7 present the block diagram of the AVPIT at the reception of both Interest and Data packets, respectively.

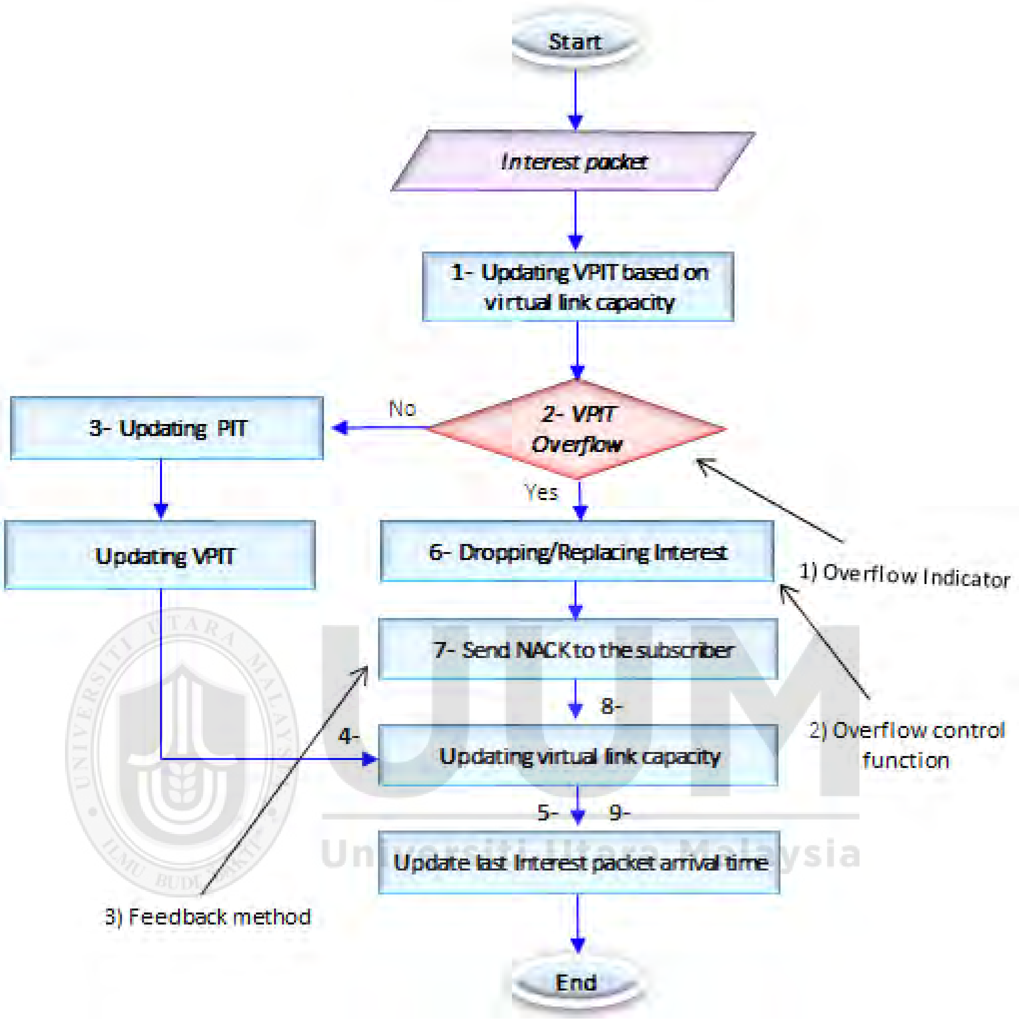


Figure 3.6. AVPIT: Reception Interest Block Diagram

The proceeding mechanism of an AVPIT at the reception of Interest packets and actions that indicates the PIT overflow is as follows:

- For each incoming Interest packet to the PIT.

i. The VPIT is updated based on the virtual link capacity.

ii. VPIT is checked whether it is overflowed or not.

iii. In case of the VPIT is not overflowing, PIT is updated by increasing it, VPIT size is updated by increasing it as well.
iv. Virtual link capacity is calculated and updated.
v. The last step in this case is updating the last Interest packet arrival time.
vi. Otherwise, the VPIT is overflowing. The new Interest packet should be dropped from VPIT, and the real Interest packet is replaced with exit entry inside the PIT.
vii. NACK packet is sent to the subscriber nodes to inform them about the PIT status in order to decrease the Interest packet sent rate.
viii. Virtual link capacity is calculated and updated.
ix. The last step in this procedure is updating the last Interest packet arrival time.

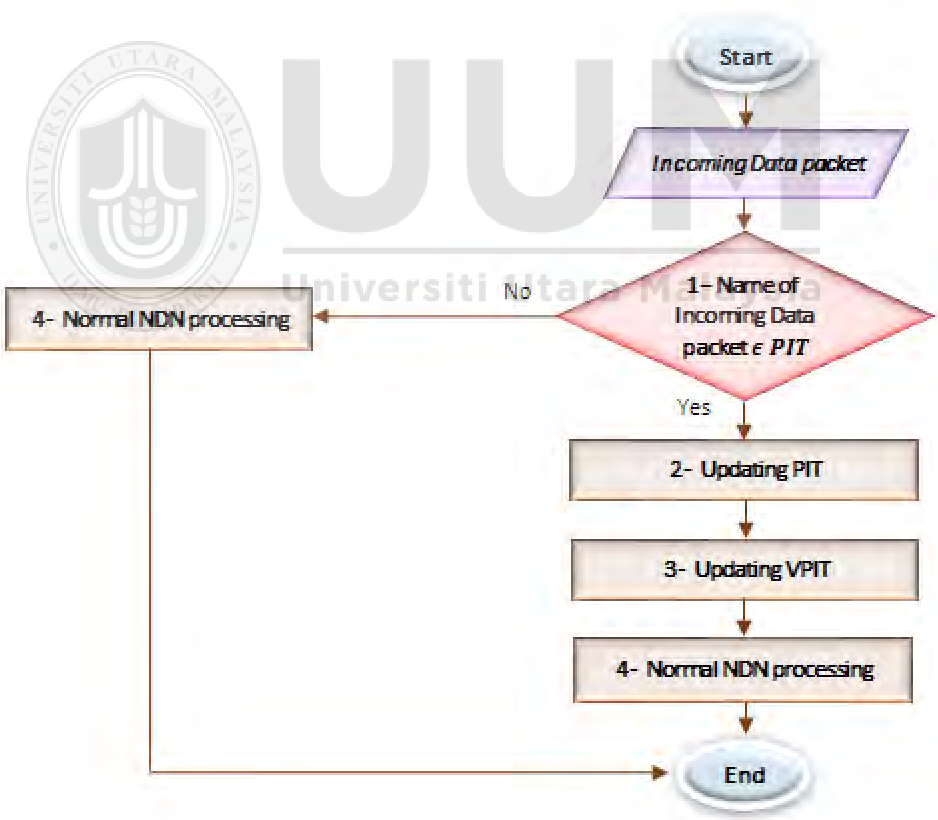


Figure 3.7. AVPIT: Reception Data Block Diagram

- For each incoming Data packet to the NDN router.

i. Verifying the name of the available Data packet with PIT entries. If a match is found, the entry should be deleted from the PIT.
ii. The PIT is updated by decreasing it.
iii. The VPIT is updated by decreasing it.
iv. Continuing with normal NDN processing.
v. Otherwise, normal NDN processing.

To validate and evaluate the accuracy of the proposed AVPIT, a Dumbbell and Tree topologies are considered with a variety of scenarios using the ndnSIM simulator. AVPIT was validated and evaluated to ensure that it meets the intended requirements in terms of PIT length, and Interests drop based on NDN router. AVPIT validation focuses on considering the relationship between the PIT overflow and the number of PIT entries. In addition, for evaluating the AVPIT, the obtained results using AVPIT were compared with the traditional NDN PIT [1] that already was implemented.

3.4.3 Smart Threshold Interest Lifetime Mechanism

The main goal of Smart Threshold Interest Lifetime (STIL) mechanism was proposed to manage the Interest lifetime at content PIT in an NDN router that adjusts the Interest packet lifetimes as a function of the network load. Thus, NDN router grants larger lifetime values for incoming Interest packets in case of low traffic and when the PIT is not full. On the contrary, NDN router grants shorter lifetime values for incoming Interest packets in order to avoid overstaying of PIT entries during heavy network load and the PIT is full. Figure 3.8 illustrates the block diagram of the STIL mechanism at the reception of Interest packets and treats the Interest lifetime.

[1] Note that the "NDN PIT" in this thesis refers to the standard PIT in NDN router without any additional enhancement [214, 215]

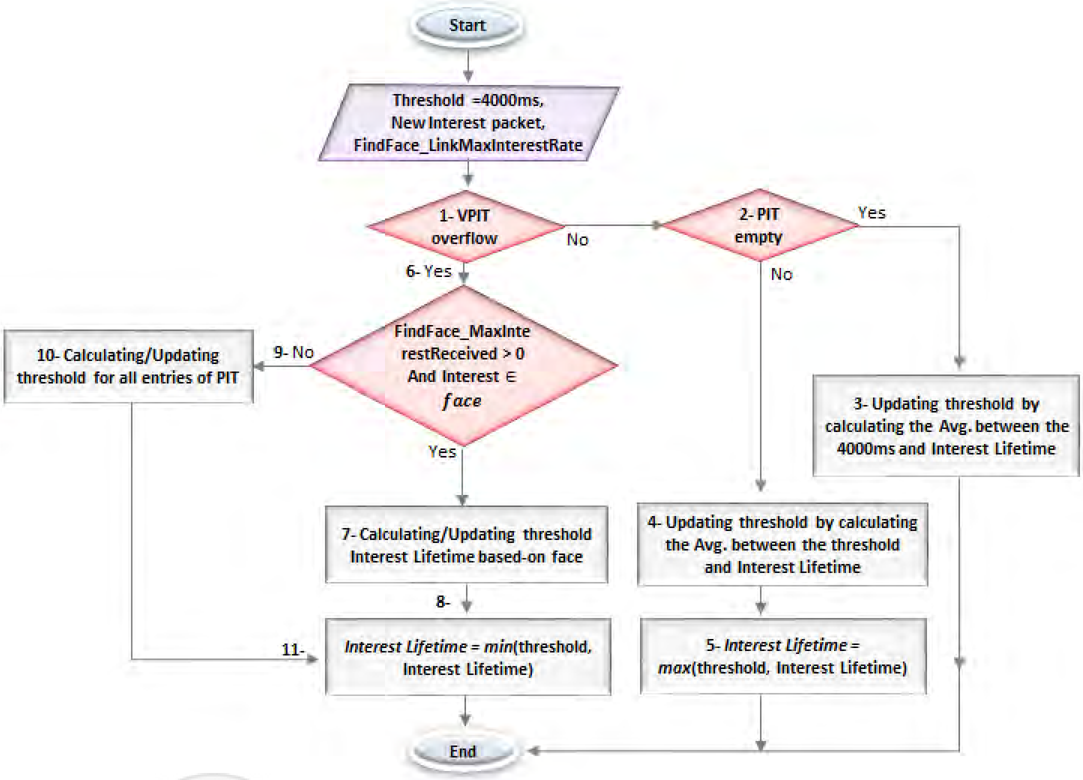


Figure 3.8. STIL Block Diagram

- The summary of this mechanism is listed as follows:

i. If the VPIT is not overflowing.

ii. PIT is checked whether it is empty or not.

iii. If the PIT is empty and the incoming Interest packet is the first arrival of the PIT, the threshold lifetime is updated by calculating the average Interest lifetime between the first incoming Interest packet and the initial value of threshold lifetime.

iv. Otherwise, the threshold lifetime is updated by calculating the average Interest lifetime between the incoming Interest packet and the threshold lifetime.

v. Threshold is checked whether its large than Interest lifetime, then Interest lifetime is updated.

vi. On the other case, if the VPIT overflow, checking whether the number of Interest packet received via the NDN router's faces is similar or diverse.

vii. If the result implies that the the number of Interest packet received is diverse. Hence, it checks whether this packet came from the same face that has the maximum number of Interest packet received. If it is from the same face, then the threshold Interest lifetime is updated by calculating the average Interest lifetime for only the Interest that has come from this face inside the PIT.

viii. Threshold is checked whether it is less than Interest lifetime, then Interest lifetime is updated.

ix. Otherwise, the threshold lifetime is updated by calculating the average Interest lifetime for all Interests included in the PIT.

x. If the result implies that the Interest packet received is the same number of Interest packet via all faces. In such a case, the threshold Interest lifetime is updated by calculating the average Interest lifetime for all Interests included in the PIT.

xi. Threshold is checked whether it is less than Interest lifetime, then Interest lifetime is updated.

To validate and evaluate the accuracy of the proposed STIL, Tree and Abilene topologies are considered with a variety of scenarios using the ndnSIM simulator. STIL was validated and evaluated to ensure that it meets the intended requirements in terms of average Interests packet lifetime, Interest packet retransmission and Interest stratification rate based on NDN router. STIL validation and evaluation considers the relationship between the PIT overflow and the Interest packet lifetime value. The obtained results (using STIL) were compared with the NDN PIT and DPEL solutions.

3.4.4 Highest Lifetime Least Request Policy

The replacement policies are helpful in evicting the entry from the PIT and build a new space for the incoming Interest packet. Because of the limited size of the PIT, a PIT cannot store the entries requests when this table becomes overflow. Hence, the replacement policy was used to determine PIT entry to be changed for efficient PIT utilization. Highest Lifetime Least Request (HLLR) policy is a type of memory management algorithm which is used to manage memory within the NDN routers when the VPIT indicates that the impending overflow of the PIT.

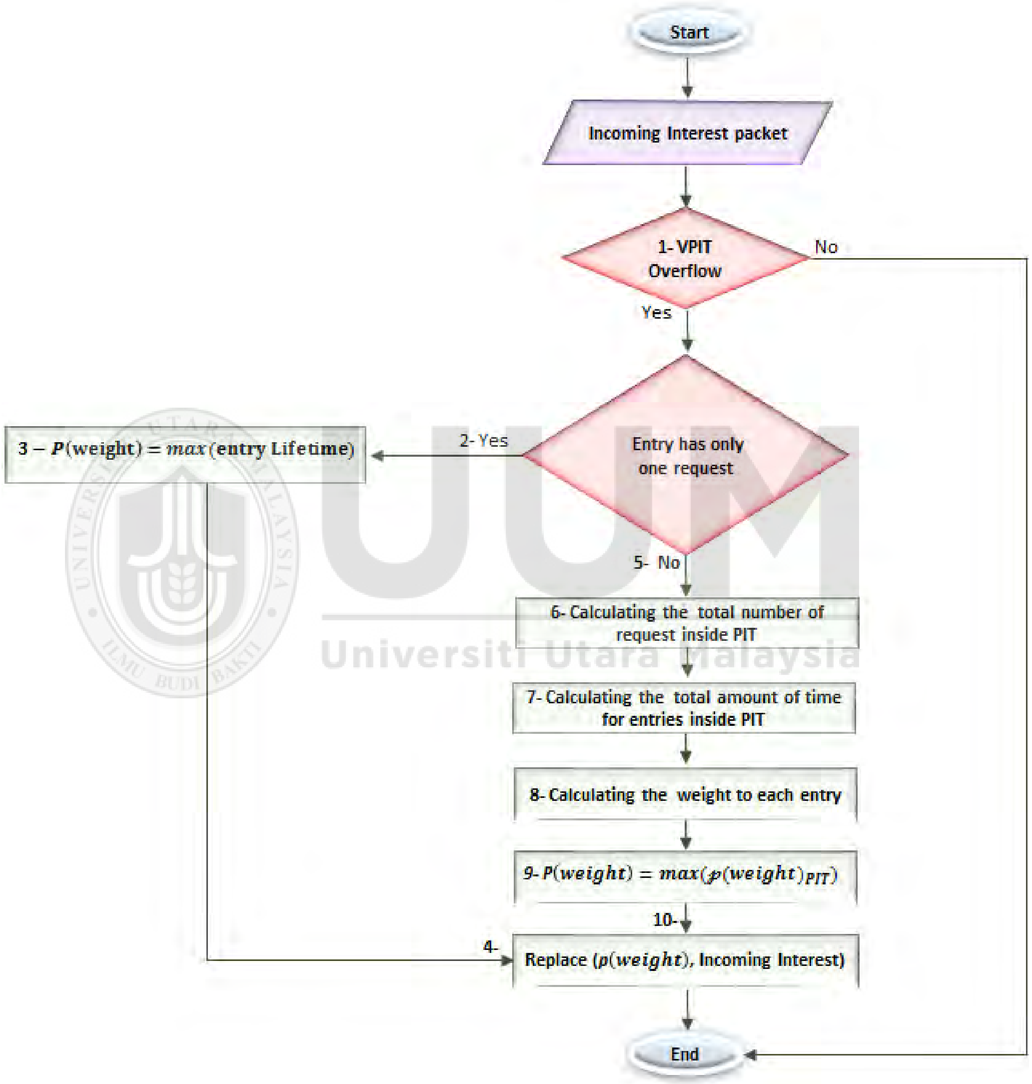


Figure 3.9. HLLR Block Diagram

The traditional characteristics of this policy allows the PIT to keep track of the number of times an entry packet is requested from memory using a counter to count every face that requests the this entry from the PIT. Each time a reference is made to that entry, the counter is increased by one. This algorithm is used to evict determine entry from the PIT and create a space for the new entry to be stored. In this section, the block diagram of the HLLR policy is presented at the reception of Interest packets and their replacement as well, as showed in Figure 3.9 above.

- The summary of this policy is outlined below:

i. As the VPIT overflows.
ii. Firstly it checks the PIT, finding the all entries that has requested only one time.
iii. Entry weight *(*i.e., the weight needed in order to determine which entry should be evicted from PIT) is calculated by finding the entry that has maximum lifetime.
iv. This entry is replaced with the incoming Interest.
v. Otherwise, the entry weight is calculated based on Interest lifetime and number of request for each entry inside PIT.
vi. The total number of request is calculated for all entries inside PIT.
vii. The total amount of entries lifetime is calculated for all entries inside PIT.
viii. The weight to each entry is calculated.
ix. The entry which has the maximum weight is determined.
x. This entry is replaced with the incoming Interest.

To validate and evaluate the accuracy of the proposed HLLR, Tree and Abilene topologies are considered with a variety of scenarios using the ndnSIM model. HLLR was validated to ensure that it meets the intended requirements in terms of Interests retransmission, Interest Drop and Interest satisfaction based on NDN router. HLLR validation focuses on considering the relationship between the PIT overflow and the number of

PIT entries. The obtained results using HLLR were compared with the available PIT replacement policies (e.g., Persistent, Random and LRU).

3.5 Design of Experiments

The proposed approach implements and verifies within the environment of a simulation tool. The simulation for generating traffic Interest was used for validation purposes. Therefore, an adequate approach in implementing any proposition is to adopt a reliable, verified, and scholarly approved simulation program. Hence, this section focuses on editor language, common simulators of the ICN and/or NDN architectures, simulation setting and networking topologies based on literatures.

3.5.1 Editor Language

In this research, the proposed approach was transformed into C++ code since ndnSIM requires C++ is the base programming language. In addition, all models should be verified to assure that they have been coded properly and free of bugs or errors. Therefore, Eclipse (see [301]) C/C++ Development Tool (CDT) that runs on top of the Eclipse platform was used for this purpose. The Eclipse IDE for C/C++ Developers provides an advanced functionality for C/C++ developers, including an editor, debugger, launcher, parser, and makes file generator, as shown in Figure 3.10.

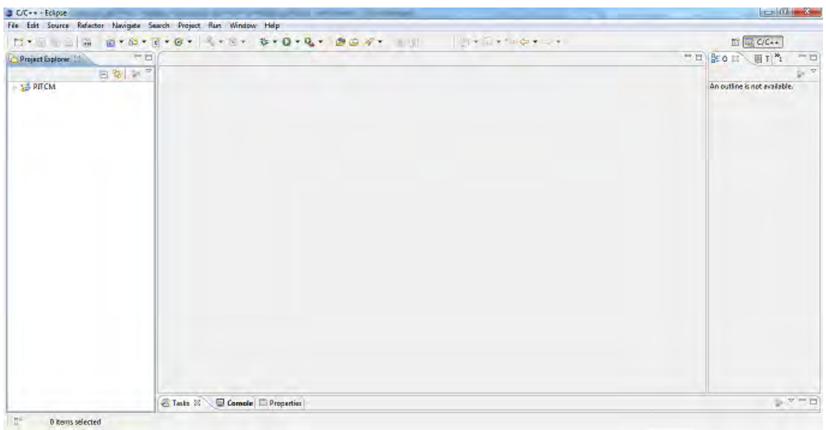


Figure 3.10. Eclipse C/C++ Development Tools [301]

Moreover, CDT's Code Analysis (CodAn) integrated in Eclipse is capable of assisting the researcher by highlighting and indicating possible syntax errors to the researcher as he or she types the code, finds bugs, and other issues as problems, warnings, and so on. CodAn works by scanning C++ code and checks for potential programming problems as well as syntax and semantic errors, as shown in Figure 3.11.

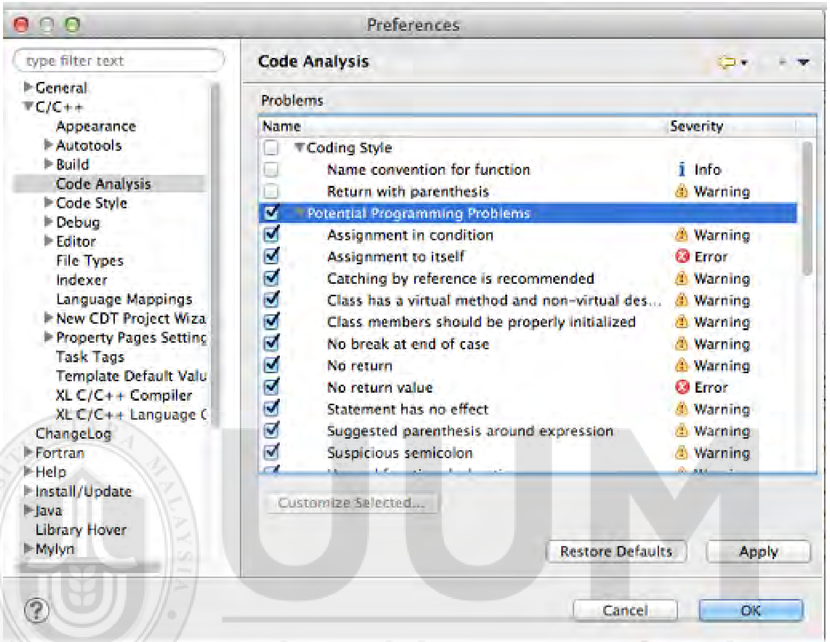


Figure 3.11. CDT's Code Analysis in Eclipse Model

3.5.2 Simulation Settings

All experiments that are presented in Chapters 4 and 5 were performed using the ndnSIM. A comprehensive simulation is performed particularly by measuring the many performance metrics as presented in Section 3.7.1 via using a machine with Linux Ubuntu 14.04 as operating system, because the ndnSIM works with high efficiency in Linux environment than other operating system. And the hardware computer Intel Core (TM) i7-3612QM at 2.10 GHz CPU, 16 GBytes of DDR3 RAM.

For more illustrations, Table 3.1 is described the main notations that are used in Chapter Four.

Table 3.1

Summary of the Notation Used in this Thesis

Symbol	**Description**
N	Set of NDN nodes
n	NDN node
l	Link
η_L	Number of links connected at NDN router
m_n	Interest rate
$\check{C}$	Virtual capacity
C	Real capacity
f	Faces
f_r	Face requests ratios
λ_{PIT}^{in}	Interest packet flow into to the PIT
λ_{CS}^{in}	Interest packet flow into the CS
λ_{CS}^{out}	Interest packets flow that are not satisfied by a Data packet
T_{th}	Threshold lifetime of PIT
p	Interest/Entry
p_τ	Entry lifetime
R	Set of requests ratios for all faces
r	Request
η_{face}	Number of faces at NDN router
η_r	Total number of request received from given face
η_R	Total number of request received from all faces
λ_{CS}^{hit}	Content Store hit probability
λ	Interest packet arrival rate at the face (f)
Q_τ	Total amount of lifetime for all entries
η_{PIT}	Limit size of PIT
PIT_{size}	Number of entries currently in the real PIT
$VPIT_{size}$	Number of entries currently in the VPIT
t	Process time
s	Arrival time of previous Interest packet
h	Current time
f_{max}	Face number that has received maximum Interest rate
w	Entries belong face (f) into PIT
r_c	Frequency of the entry already caching in the PIT
μ_τ	Maximum PIT entry lifetime

In addition, the full range of parameters that might have impacts on the experiment, must be opted precisely for increasing the validity of simulation results. Hence, Table 3.2 is illustrated the quantity's parameters that chosen in the present simulation scenarios with their design ranges.

Table 3.2
Design Range for the Simulation Parameters

Parameter	Description
Simulation Environment	ndnSIM [47, 207]
Simulation Topology	Abilene [224], Rocketfuel-mapped AT&T [234]
Forward Strategy	Flooding Strategy [47]
CS Replacement Policy	LRU [234]
PIT Replacement Policy	LRU, Random, Persistent [47, 207]
Interest Rates	1000 - 10000 (Interest/second) [233, 170, 52]
Content Store	10000 Data packets
PIT Size	1000 - 10000 Interest packets
Interest lifetime	4000 ms [69, 302]

In this research, network topologies with different network sizes and varying Interest traffic load were used to test and evaluate the proposed models. For this purpose, two enterprise topologies were used, which are Abilene topology and Rocketfuel-mapped. The experiment outcome for each simulation scenario compares the related work simulation results with the PITCM.

3.6 Verification and Validation

Two steps exist for ensuring the quality of a simulation study, namely Verification and Validation (V&V). Model verification verifies that the model is transmuted from one form into another as required with adequate precision. Model verification deals with developing the accurate model. The precision of transmuting a problem formulation into a model specification or the precision of transforming a model illustration from a micro flowchart form into an executable code is estimated in the model verification [303]. In other words, verification is to make sure that the model implements those assumptions correctly. ndnSIM environment was used in this research. Hence, ndnSIM requires C++ as the base programming language. Therefore, all proposed components of PITCM are transformed into C++ code. AVPIT, STIL, and HLLR must be verified to ensure that the code is correct and free of errors.

The second step called validation is to make sure that the assumptions are realistic. Model validation can be defined as "*substantiating that the model, within its domain of applicability, behaves with satisfactory accuracy consistent with the study objectives. Model validation deals with building the right model*" [304]. Validation must be performed to make sure that the mechanisms meet the requirements in terms of the techniques that are employed as well as the results that are obtained, which is included in the task to construct the precise model.

According to Balci in [303], V & V techniques can be classified into four categories: "informal, static, dynamic, and formal". The primary approach applied in this research is a dynamic technique that is commonly used in model V & V. Hence, it is used to ensure that the implementations of the mechanisms are correct by comparing testing that is executed under different conditions and the obtained values. In other words, dynamic techniques, such as Graphical comparison, Performance testing and Sensitivity analysis that are required proposal model execution and are intended for evaluating the proposal model based on its behavior of execution [305]. Brief implications of verification and validation process are demonstrated in Figure 3.12.

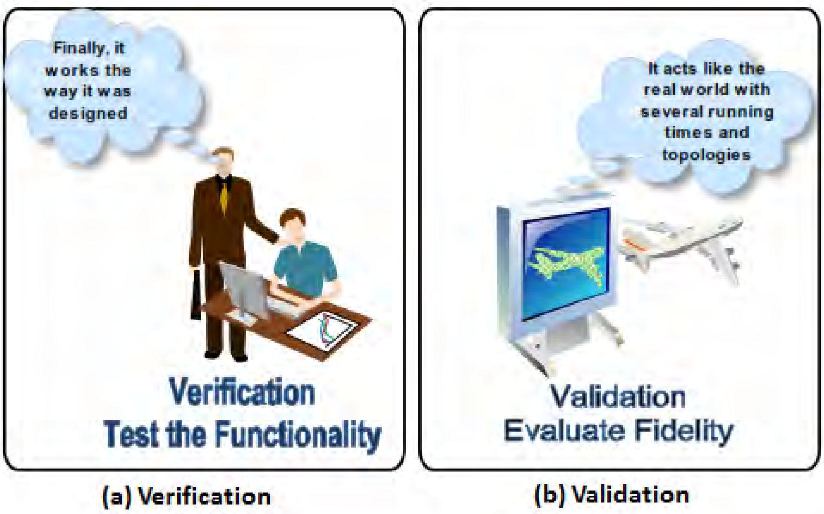


Figure 3.12. Verification and Validation Process [198]

Graphical comparison technique is an inelegant, heuristic and quite practical approach which is useful as a preliminary approach to model V & V. The graphs of results that generated from simulating the model over time were compared with the graphs of results of valid mechanisms variables to investigate behaviors i. e., similarities in periodicities, number and location of inflection points, trend lines, skewness, rise linearity, phase shift, and exponential growth constants. Therefore, AVPIT is validated by using Graphical comparison technique in order to show a behavior of AVPIT.

Performance testing technique is used for testing whether all characteristics of the performance are measured as well as evaluated with enough accuracy, and all established performance requirements are satisfied. Thus, STIL uses performance testing technique to perform validation to ensure that the mechanism operation is in accordance with the specifications. *Sensitivity analysis* technique is performed by methodically changing the input values and parameters of the proposed model across a range of interests and taking into consideration the effect on the behavior of the model. Sensitivity analysis can determine those input values of which model behavior is extremely sensitive. Hence, the model can be improved by ensuring that those values determine with an adequate accuracy. Therefore, HLLR was validated by using graphical comparison technique in order to show a similar behavior of HLLR compared to the behavior of Persistent policy.

Accordingly, as clarified in Chapter One, this research focuses on the design and implementation of PITCM that addresses the overflow problem of PIT in Named Data Network architecture. Hence, the research has proposed simulation models to verify the results. Therefore, PITCM has been implemented in ndnSIM environment, and the results were obtained from other research works, which can be utilized to validate the results of simulation, respectively. In order to achieve the study goal, Figure 3.13

presents a block diagram describing the main stages of the PITCM according to this research.

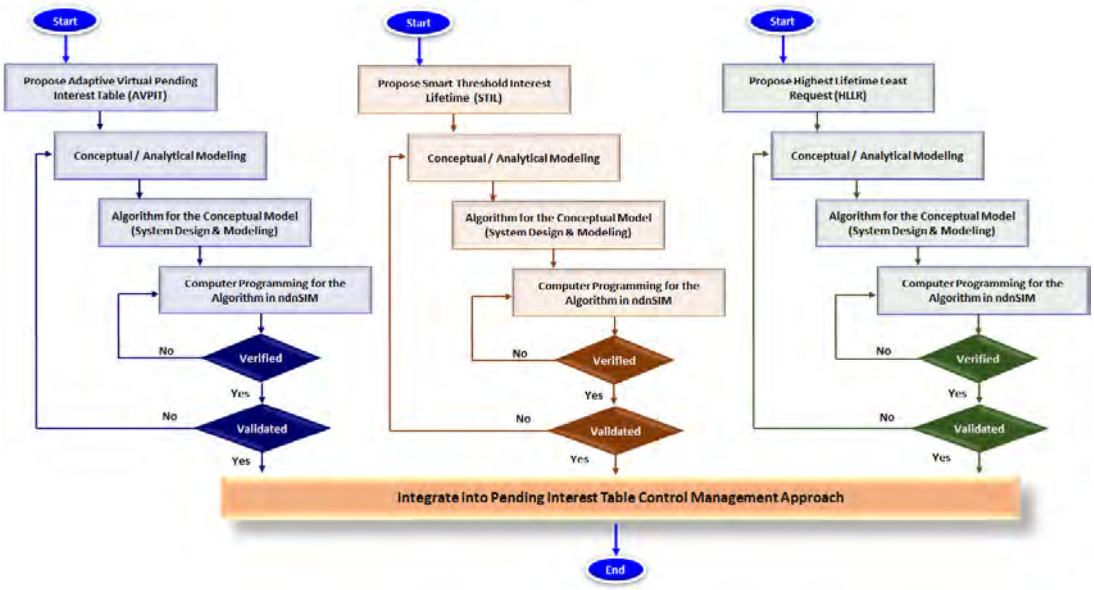


Figure 3.13. Stages of Verification and Validation

The specifications of the proposed approach are described in the first block. There are three components in this research which are AVPIT mechanism, STIL mechanism, and HLLR policy. The second block constitutes the conceptual/analytical modeling of this research, which is built to describe the expected preferred and improved situation using the proposed AVPIT, STIL and HLLR. The third and fourth blocks represent the design and implementation of the proposed AVPIT, STIL and HLLR in the simulation environment. In the next blocks, verification and validation of the computer program model are checked to make sure of their usability and performance, respectively. Finally, integration AVPIT, STIL and HLLR into PITCM approach in order to evaluation and analysis in the next stage.

3.7 Performance Evaluation

Evaluation of the performance methods are a very crucial step in evaluating the final results of any research or project [306, 307]. It is required if a system designer intends to compare a number of alternative designs for finding the accurate design [299, 308]. Accordingly, different ICN architectures were evaluated in the literature review using a mixture of theoretical analysis, empirical measurements (testbed) and simulation or emulation techniques. All popular methods were evaluated including network architectures, services and protocols in the networking community. Researchers usually follow a specific methodology based on the objectives of their experiments (e.g., to evaluate scalability, quantify resource utilization, and analyze economic incentives).

In addition, the experimental process itself in addition to the evaluation methodology are now actively investigated in ICN architectures. There are many factors that can affect the experimental results, such as network condition (e.g., Available link capacity, topology selected, link delay, node mobility, background traffic load, loss rate characteristics, disruption patterns, and the variety of devices used) [215]. This research has evaluated the AVPIT, STIL and HLLR individually by comparing with the current works in the first stage as well as with the traditional NDN PIT in the next stage after integrating these components in PITCM approach.

3.7.1 Evaluation Metrics

The main step in the performance evaluation is the performance metrics selection. According to Ghazali in [308] and Suki in [299], selecting the performance evaluation metrics is a key step and an important part in all performance evaluations. In this regard, Habbal in [306] argues "*performance metrics can mean different things to different researcher depending on the context in which it is used*". Thus, the selection of the proper performance metrics is very important to investigate the behavior of the approach of different viewpoints. The use of multiple different metrics gives us a complete picture of the performance of the proposed approach [309]. Selecting the proper performance metrics is also very important for investigating the behaviors of the protocol from different perspectives. In addition, the use of multiple different metrics gives us a complete view of the performance of the proposed mechanism.

There are four reasons for choosing the suitable metrics for performance evaluation in simulation technique. Firstly, it should be available or simple to be implemented in ndnSIM. Secondly, it should be either most recent or most famous. Thirdly, it is more preferable if the approach is utilized in real-life routers. Finally, most previous studies have used it as a reference. Hence, this study focuses on the PIT Length, Interest Retransmission Rate, Interest Drop Rate, Interest Satisfaction Delay and Interest Satisfaction Rate metrics that used to measure the performance of the proposed objectives of this research. This performance evaluation was done by means of the same metrics that were used by other researchers in previous studies.

- PIT Length refers to the amount of non-expired pending Interests that exist on each and every router. Thus, it obtains the real number of cached PIT entries [61, 278, 170, 60]. The performance of PIT is good when the number entries currently in the PIT are not being reached the maximum capacity of the PIT.
- Interest Retransmission Rate (IRR) is perceived by the subscriber that measures the percentage of Interest retransmissions among the entire outgoing Interest packets when it is reissued because of timeouts [60, 310]. The performance of PIT is good when the IRR is decreased as much as possible without effect on the network utilization.
- Interest Drop Rate (IDR) can be explained as the percentage of dropped Interest packets among the entire incoming Interest packets [170, 109]. Thus, it refers to the number of Interests drop in the PIT because of the Interest replacement or Interest lifetime has expired. The performance of PIT is good since the IDR is decreased.

- Interest Satisfaction Delay (ISD) refers to the time interval between the first Interest generated by the subscriber for the specified content and the Data packet received in satisfying [115, 214, 116, 215]. The performance of PIT is good when the ISD is minimized. This is due to the Data packet not forwarded towards the longer paths.
- Interest Satisfaction Rate (ISR) refers to the percentage of satisfied Interests per NDN router in the network [61, 311, 38]. The performance of PIT is good as the ISR is high; from ISR, it is possible to obtain the effective amount of Data packet that satisfied by the NDN router—how much data effectively reached the subscribers.

3.7.2 Results Analysis and Discussion

There are two sources of the results. Firstly, results are accumulated from the simulation or implementation of the proposed mechanism and comparison with the related works (see Chapter Four), and secondly, results are collected from the evaluation of PITCM and comparison with the previous work (see Chapter Five). Both results necessitate multiple analyses which vary and may be similar to each other in some aspects. The former experiments which are related to the implementation of the proposed mechanism require a specific data analysis related to the performance. By comparison, data analysis for evaluation of the mechanism takes a different approach. The evaluation of the proposed approach is achieved by its results with others' current works using specific metrics, such as PIT Length, Interest Retransmission Rate, Interest Drop Rate, Interest Satisfaction Delay and Interest Satisfaction Rate.

3.8 Research Conclusion

The previous section shows the final stage of developing the PITCM which is the main goal related to this study. This section demonstrates the summary of the research,

research contributions, research limitations, and the recommendations for future works that is presented in Chapter Six.

3.9 Research Reporting

A research report is a document prepared by a researcher or a group of researchers who focus on a specific study (field). The work of a researcher becomes gratifying only if the report of the discoveries becomes available to the users who are able to utilize it. It is also important that the report is operational, which can mean different things for various types of users. Hence, this study is reported in six chapters, which are the introduction, the literature review, the research methodology, the design and implementation of PITCM, the evaluation and a conclusion. This research focuses on the PIT in NDN architecture to address the effect increasing the network load on the performance of PIT, which is covered from Chapter One to Chapter Six. Hence, this research does not provide a detailed survey, but the sufficient amount of information that can work for this study is presented comprehensively.

3.10 Summary

This chapter has presented in general the details about the methodology and research design that were used to achieve the research objectives. A specific methodology was used to realize all the research steps starting from the area of understanding the literature review. Accordingly, the research purpose was identified and followed by designing and implementing Adaptive Virtual Pending Interest Table (AVPIT) mechanism, Smart Threshold Interest Lifetime (STIL) mechanism, and Highest Lifetime Least Request (HLLR) policy, which were integrated in Pending Interest Table Control Management approach in order to solve the problem of PIT (overflow). The experimental design, verification, validation, and evaluation were also presented in this chapter.

CHAPTER FOUR
PENDING INTEREST TABLE CONTROL MANAGEMENT APPROACH

Chapter One and Chapter Two thoroughly illustrated the background of this research through the introduction and the literature review that covered the problem definition, the research objectives, and the research scope. In addition, infrastructure building of the academic background was clarified to explain and criticize the related works, which was used to address the problem. Chapter Three established the research methodology as a guideline to achieve the objectives of this research as well as illustrated all the steps needed for performance evaluation of the developed approach.

Next, this chapter designs and implements a Pending Interest Table Control Management (PITCM) approach components, aims to enhance the management operations of Pending Interest Table (PIT) by monitoring and controlling incoming Interest packets as well as managing PIT entries in order to mitigate PIT overflow. Section 4.1 explains the Pending Interest Table Control Management approach. The Adaptive Virtual Pending Interest Table mechanism, Smart Threshold Interest lifetime mechanism and Highest Lifetime Least Request policy are covered in detail (theory, description, model analysis, verification, validation, and evaluation) in Sections 4.2, 4.3, and 4.4, respectively. Finally, Section 4.5 concludes the chapter with a summary.

4.1 Pending Interest Table Control Management Approach

The specific problem that was considered in this research, as stated earlier, was achieving a controlling and monitoring mechanism of the incoming Interest packets, and managing the entries of PIT in the Named Data Network environment. The ultimate aim of this thesis was to design and implement a Pending Interest Table Control Management (PITCM) approach in the Named Data network architecture. PITCM is a new PIT management approach based on NDN PIT in order to monitor, control, and manage the PIT, which leads to mitigate PIT overflows, thus, increasing the utilization of PIT. PITCM incorporates three proposed components, namely Adaptive Virtual Pending Interest (AVPIT) mechanism, Smart Threshold Interest Lifetime (STIL) mechanism, and Highest Lifetime Least Request (HLLR) policy. In this chapter, AVPIT, STIL, and HLLR shall be described comprehensively.

The system model in the present study must capture PIT features and concepts (see Section 2.5.1). Hence, each component of the PITCM approach was designed to accomplish a specific objective as presented in Chapter One. In the next section, the proposed approach is presented in detail including the approach components with others necessary mechanisms for completing the approach design. According to Mamun in [312], it is important to define some of the assumptions before presenting any proposed model. Hence, the assumptions made to develop this model are outlined as follows:

- NDN nodes are homogeneous and have only one CS, FIB, PIT, and VPIT.
- Each Interest and Data packet have a constant size.
- The main cause of Interest retransmission and Interest drop is PIT overflow. The Interest retransmission and Interest drop probability due shallow or overflow of buffer is omitted.
- Overflow occurs in PIT, where the Interest packet becomes a high rate without satisfied by CS.
- The main cause of Interest retransmission and Interest drop is PIT overflow.
- NDN nodes are classified into three types: router nodes (intermediates), subscriber nodes (consumers/requesters), and publisher nodes (services/providers).

- The initial Interest lifetime is fixed for all Interest packets; initially it is 4000 ms.
- Adopting the form of the penalty function is derived from loss probability for a M/M/1 queuing model.
- Consider which consists of a set (N) of NDN routers; where $N = \{1,2,3,...,n,...N\}$, for $n \varepsilon N$ with Interest transmission rate of m_n at time per second. Moreover, consider a node (n) contains a set (n_L) of links where $L = \{1,2,3,...,l,...,L\}$, each with finite capacity (C), and $l \varepsilon L$. In this case, each n uses a set $n_L \subseteq L$ of the links. Hence, the N set that uses a link (l) is represented as $N = (n \varepsilon N, l \varepsilon n_L)$. Let p represent an entry with entry lifetime of τ at time per millisecond (p_τ). Let $m_n = [m_1, m_2, ..., m_n]^T$ be a vector of all node Interest packet rates. Also, consider PIT_{size} to be a fraction of the number of entries currently in the PIT, since $PIT_{size} = \{1,2,3,.....,p,......PIT_{size}\}$.

4.2 Adaptive Virtual Pending Interest Table Mechanism

NDN routers need to define efficient mechanisms for traffic controlling when numerous subscribers compete to access the same or different resources, which may lead the PIT to be flooded. As a result, Interest drop, Interest retransmission, or Interest satisfaction delay would be increased. This section provides a particular mechanism, namely Adaptive Virtual Pending Interest Table (AVPIT) for detecting the impending PIT overflow. In addition, the proposed mechanism works as a response decision to handle the PIT overflow recovery.

In order to have the content-centric concept attained, each NDN router needs to be outfitted with four components, which are Content Store (CS), Forwarding Information Base (FIB), Pending Interest Table (PIT), and Virtual Pending Interest Table (VPIT). Since the VPIT is a logical component with a buffer size equivalent to the buffer size of the actual PIT, the VPIT is used as an indicator in order to predict the impending overflow of the PIT. This section presents an AVPIT model, the theory, AVPIT analysis, validation, and evaluation of the proposed mechanism.

4.2.1 Description of AVPIT Mechanism

The AVPIT mechanism was designed based on the AVQ-AQM scheme that attempts to maintain the Interest arrival rate at PIT. The motivation behind the AVPIT mechanism was to design an efficient PIT management mechanism in low-delay, low-drop, and high utilization of the PIT. An appealing feature of the AVPIT mechanism is that the mechanism uses a virtual PIT to make the Interest arrival rates adhere to the desired utilization of the link. The AVPIT mechanism maintains a virtual PIT, where the capacity is known as the virtual link capacity, which is less than the real link capacity. As an Interest packet is received in the real PIT, the virtual PIT is also updated to reflect the new Interest packet arrival.

When the virtual PIT overflows, Interest packets in the real PIT are replaced, whereas the fictitious Interest packet (i.e., size of the packet) in the virtual PIT is discarded. In addition, AVPIT mechanism reduces the virtual capacity whenever the total Interest packet flow into the link exceeds the desired utilization at the link. If the total Interest packet flow into the link is smaller than the desired utilization at the link, the virtual capacity increases. Thus, by reducing the number of drops or replacing Interest packets, this allows subscribers to increase their Interest packet rates.

This mechanism basically utilizes the Interest packet rate mismatch and regulates utilization rather than just based in the PIT length. Thus, AVPIT uses both the VPIT size and the Interest packet arrival rate as part of its overflow indicator and attempts to keep the Interest arrival rate and PIT size around their respective individual target values.

AVPIT then predicts the impending overflow of the PIT when the number of entries grows due to the high packet arrival rate at the PIT. Based on this, this mechanism answer of the first question of the research questions that highlight the direct influence workload on the PIT and network performance as well. Therefore, the main goals of AVPIT are:

- to make the virtual capacity adaptive at each Interest packet arrival epoch,
- to react to impending PIT overflow before it takes place, and
- to reduce waiting time for subscribers/download nodes by informing them faster than Interest packet timeout, as well as to reduce the effects of Interest packet retransmission for NDN routers.

Example: the PIT overflow would occur when the total amount of incoming Interest packets to the NDN router exceeds the PIT capacity. During the period of heavy rate with delay from transmitters in the network, the PIT in the NDN routers is filled. If this delay continues, the PIT would then be overflowed. Thus, the Interest packets which cannot be accommodated in the PIT are discarded by the NDN router, and the Interest packet must be retransmitted by the subscribers. As a result, dropping Interest packets by the NDN router would clearly indicate PIT overflow. To illustrate the AVPIT design, a simple example is shown in Figure 4.1 and Figure 4.2.

In the given example, the NDN router comprises four components (i.e., CS, FIB, PIT, and VPIT). For instance, it is assumed that the PIT can cache eight Interest packets in total, whereas the size of VPIT is the same as that of the PIT (see Figure 4.1-a). Upon each Interest packet arrival, a fictitious Interest packet enters the VPIT if there is sufficient space in the PIT, while the real packet enters into the PIT (see Figure 4.1-b).

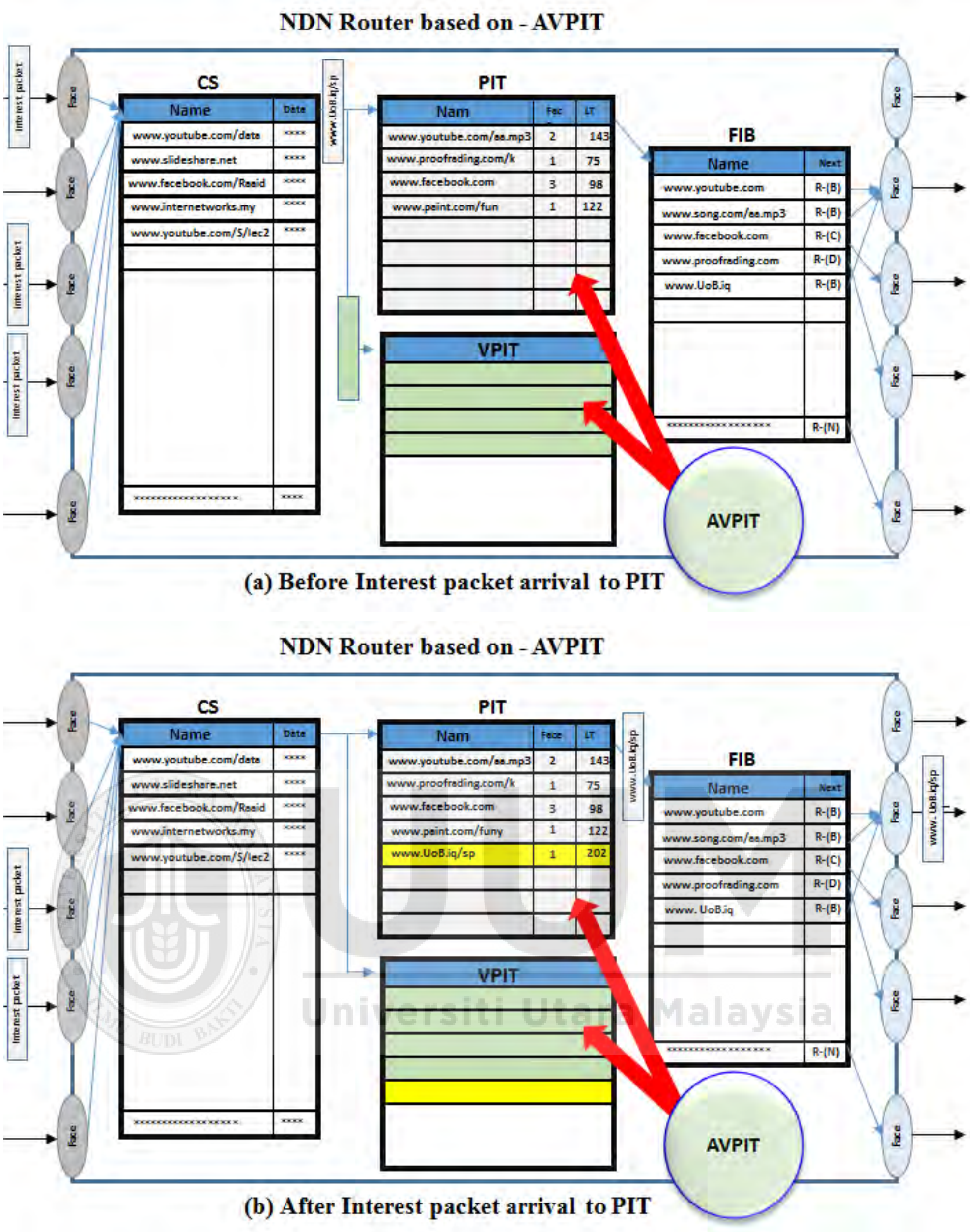


Figure 4.1. AVPIT: When VPIT is not Overflow

The VPIT is updated based on "update virtual link capacity" each time the NDN router receives a new Interest packet (see Figure 4.2-a). Upon the arrival of the new Interest packet, if the VPIT overflows, the Interest packet is discarded in the VPIT and the real Interest packet is replaced with the evicted entry from the PIT. At the same time, the NACK packet notification is sent to the downstream nodes (see Figure 4.2-b).

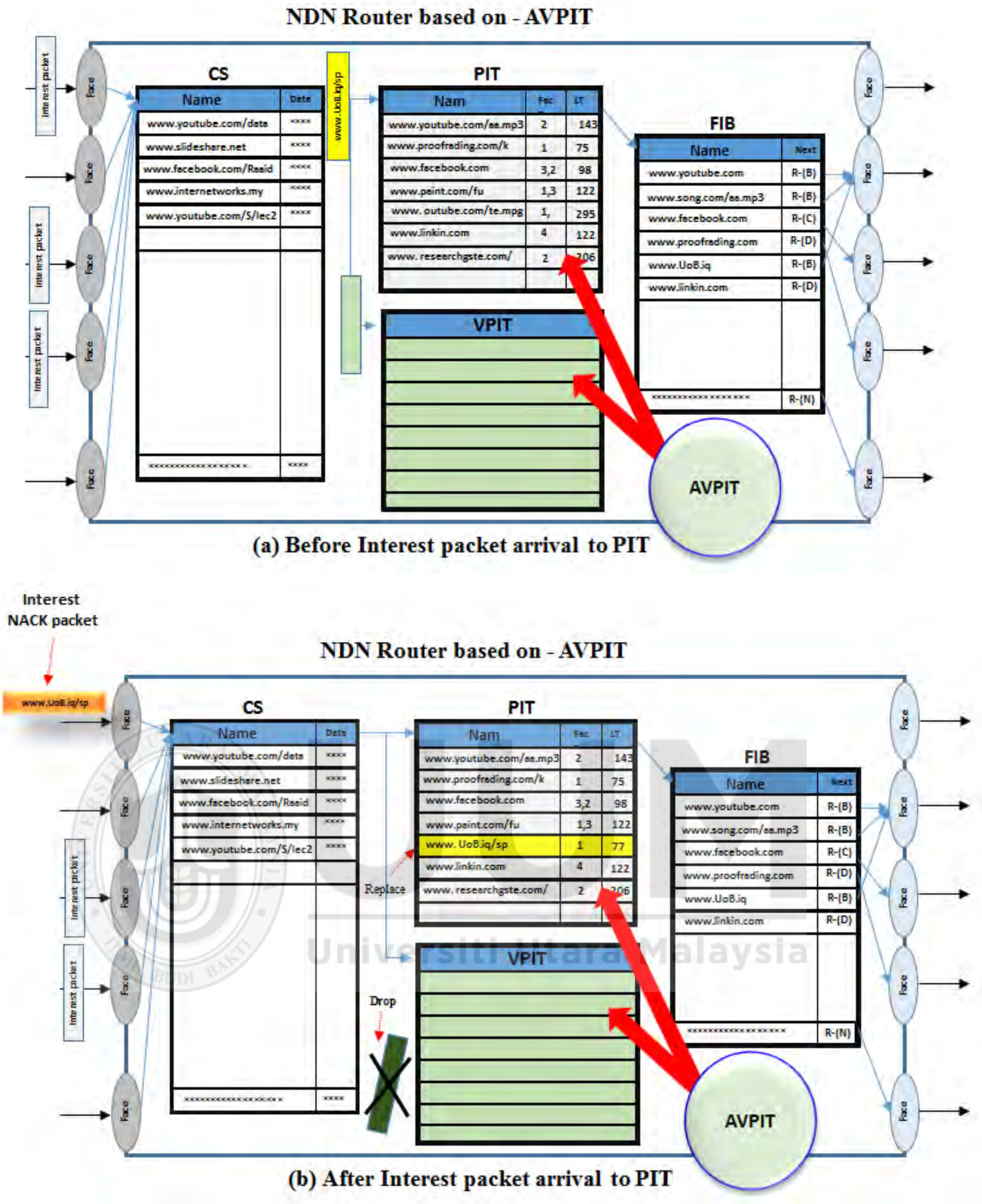


Figure 4.2. AVPIT: When VPIT is Overflow

As mentioned earlier, two operations will be active after it has been indicated that the VPIT is overflowed, which are the replacement policy and feedback operation which return the NACK packet to the downstream nodes. In the AVPIT mechanism, the replacement policy is one of the significant features which determines the efficiency of a cache. A common NDN router uses the replacement policy in cache memory management when the PIT is full. Therefore, the replacement policy can have a significant influence on PIT performance. More detail about this replacement policy is provided in Section 4.4.

Meanwhile, the Interest Negative ACKnowledgement (NACK) is a packet sent by upstream nodes to inform the downstream nodes that data cannot be retrieved or delivered in response to a request. NACK is useful in forwarding strategy as an explicit signal. In the AVPIT mechanism, Interest Negative ACKnowledgement method is adopted to notify the downstream nodes when the VPIT overflows. If the VPIT overflows, the NDN router will send the NACK packets based on statistical probabilities from which the Interest packets have been received. The possibility of sending a NACK packet also grows with the growth of the PIT entries.

The NACK packet follows the reverse path of the Interest packet by consuming the Interest information in the PIT. Subscribers expand the "Interest sending window" by one with no timeout of Interest packet. If a NACK packet is received, the subscribers maintain a rate limit of Interest sending by half (*"use an Additive Increase Multiplicative Decrease (AIMD) algorithm* [313, 227]*"*) without retransmitting the Interest packets for the NACK packets. AIMD is a congestion control mechanism used to adapt a subscriber's Interest packet request rate; Interests packet exceeding the fair share are discarded with NACK packets and simple AIMD mechanism utilized to adjust the subscriber's Interest packet request rate [227].

Yi *et al.* in [314] and Cheng *et al.* in [277, 314] proposed the Interest NACK packet as a fast failure detection and evaluated its benefits in handling network failures. Interest NACK packets may be classified under the title "*explicit feedback congestion control mechanism*" [5]. The purpose of this categorization is the definition of NACK packet itself. If an NDN router neither satisfies nor forwards an Interest packet because of link congestion, PIT overflow, and Interest loop, the NDN router sends the Interest NACK packet back to the downstream node. Nevertheless, this may create a loop in the network due to Interest packet retransmission. To overcome this issue, Ndikumana and Hong in [315] argued that once an upstream node fails to forward the Interest packet, it needs to send back the Interest NACK packet to the downstream node due to employing its own alternative paths. If the Interest packet forwarding failure continues, the downstream node repeats the same process until the Interest NACK packet reaches the subscriber. Therefore, in this research, this concept was adopted for fast network fault detection and notification.

The NACK packet must carry the same name as it appears in the Interest packet. The Interest NACK has three important parts, namely Interest Name (failed Interest), Nonce (the nonce carried), and Error code. The Error code (NACK Loop, NACK_Congestion, or NACK Giveup_PIT) is associated with the description that explains the reason why the Interest packet cannot be forwarded. Thus, proper action needs to be taken accordingly to select an alternative path [278, 315]. Therefore, returning Interest NACK packets bring three features to the system, *(i)* it cleans the PIT state much faster than waiting for timeout of Interest packet lifetime, *(ii)* the downstream nodes can identify the specific cause of a NACK in order to take an informed local recovery action [314], *and (iii)* it can help mitigate the effects of Interest flooding attacks [132].

4.2.2 Analysis of AVPIT Mechanism

The PIT is considered the most crucial component in the NDN node because it keeps track of Interest packets forwarded upstream towards a content publisher node; therefore, the returned Data packet may be sent downstream to its subscriber nodes. Consequently, a flexible mechanism is needed in order to manage the PIT when the PIT is full. Hence, a flexible mechanism would decrease Interest packet drop, and as a result decrease the Interest packet retransmission and Interest satisfaction delay. An AVPIT mechanism was proposed to manage the PIT in the NDN router for early detection of overflows.

AVPIT was designed based on Kunniyur and Srikant's scheme [191]. In the AVPIT mechanism, the link capacity of the VPIT is denoted by $(\check{C})$, where (C) is considered as the link capacity of the PIT, such that $\check{C} \leq C$. In addition, the size of the VPIT$(VPIT_{size})$ is equivalent to the size of the real PIT (PIT_{size}) (i.e., number of entries currently in the real PIT), which has a limit size (η_{PIT}). The service rate of the PIT is greater than that of the VPIT, and both have similar arrival rates. Consequently, the VPIT will boost faster than the PIT and would tend to overflow more rapidly. At any time the VPIT becomes overflowed, the Interest packet is discarded from the VPIT and the real Interest packet is replaced in the PIT. At each Interest packet arrival epoch, the virtual link capacity is updated as follows:

$$\check{C} = C - \lambda_{PIT}^{in} \tag{4.1}$$

where λ_{PIT}^{in} is the Interest packets arrival epoch into the link that has been recorded for the first time at PIT and after which forwarded them by looking up the names in its FIB. In order to simplify the equation 4.1, needs to calculate differential equations. If the NDN router receives an Interest packet, it is first checked in the CS. Let the (P_{CS}^{hit}) denote the hit probability of CS, which is a fraction of Interest packets that are satisfied by Data packets cached in the CS. Hence, there is no need to forward the Interest packet to the PIT, as mentioned in [316]. In this case P_{CS}^{hit}=1, which gives rise to the following:

$$\lambda_{CS}^{in} = \sum_{i\varepsilon|\eta_{face}} \lambda_i.(1 - p_{CS}^{hit}) \tag{4.2}$$

where λ_{CS}^{in} is the total Interest packet flow into the CS, (λ) is the Interest packet arrival rate at the face (f), and $i = \{1,2,3,...,\eta_{face}\} \ \forall \ i\varepsilon\eta_{face}$. The rate of Interest packets is forwarded to the PIT if it is not satisfied by the CS in NDN, i.e., at P_{CS}^{hit}=0, it is as follows:

$$\lambda_{CS}^{out} = \sum_{i\varepsilon|\eta_{face}} \lambda_i.(1 - p_{CS}^{hit}) \tag{4.3}$$

where $i = \{1,2,3,...,\eta_{face}\} \ \forall \ i\varepsilon\eta_{face}$.

Then, the PIT considers if no data packet have been cached in the CS. Hence, the PIT Interest is forwarded to the PIT as mentioned in [317]:

$$\lambda_{PIT}^{in} = \lambda_{CS}^{out} \tag{4.4}$$

Hence, by substitution:

$$\lambda_{PIT}^{in} = \sum_{i\varepsilon|\eta_{face}} \lambda_i.(1 - p_{CS}^{hit}) \tag{4.5}$$

where $i = \{1,2,3,.........,\eta_{face}\} \ \forall \ i\varepsilon\eta_{face}$.

For each Interest packet arrival, a false Interest packet is updated in the VPIT when there is enough space in it, and allows it to be serviced in the PIT. At any time the VPIT becomes overflowed, the Interest packet is discarded from the VPIT, and simultaneously the real packet is replaced in the PIT. This is considered as the beginning of the analysis of the AVPIT mechanism at the reception of both Interest and Data packets. As an Interest packet arrives, it considers three cases based on the relationship between the PIT and VPIT, as well as the virtual capacity of the link.

- ***Case 1***: Updated the VPIT $(VPIT_{size})$.

Let (t) denote the process time (i.e., the actual time between the Interest arrival to VPIT and current time), (s) denote the VPIT arrival time of previous Interest packet, and (h) denote the current time, so t is defined as follows:

$$t = h - s \tag{4.6}$$

An accurate t estimation is necessary to identify the changing virtual capacity link of each Interest packet, both incoming or outgoing from the system. Therefore, updating the VPIT $(VPIT_{size})$ based on the virtual capacity link $(\check{C})$ and the actual time of the incoming Interest that is needed, is estimated as follows:

$$VPIT_{size} = [VPIT_{size} - \check{C}.t]^{+} \tag{4.7}$$

Meanwhile, $[w]^{+} = max(w, 0)$ in order to ensure that the value of $VPIT_{size}$ would not be less than zero.

- ***Case 2***: $VPIT_{size} + p \leq \eta_{PIT}$

Based on the PIT concepts, if it is supposed that the current Interest packet is p, then the p is inserted into the PIT and PIT_{size} will be updated based on Equation(4.8).

$$PIT_{size} = PIT_{size} + p \tag{4.8}$$

Next, $VPIT_{size}$ is updated as mentioned in Equation (4.9).

$$VPIT_{size} = VPIT_{size} + p \tag{4.9}$$

- ***Case 3***: $VPIT_{size} + p > \eta_{PIT}$

If the incoming Interest packet overflows the VPIT, the packet is discarded in the VPIT and the real Interest packet is replaced by updating the real PIT depending on the replacement policy used by NDN router. At the same time, the related Interest NACK packet is sent to the downstream node. Following that, NDN router sends the respective Interest NACK packet to the downstream nodes to inform them that there is an impending PIT overflow in order to limit or adjust the Interest packet rate. In case 2 and case 3, when the NDN node receives the Interest packet, the link capacity ($\check{C}$) updating is derived as follows:

$$\check{C} = \check{C} + C.t \tag{4.10}$$

$$\check{C} = min(\check{C}, C) \tag{4.11}$$

$$\check{C} = max(\check{C} - p, 0) \tag{4.12}$$

The $\check{C}$ is monitored by the NDN router, which is the value of $\check{C}$ that is equal or less than the value of C. Meanwhile, when the NDN router receives a Data packet, verifies the name of available Data packet with PIT entries, and a match is found, the Data packet is then forwarded through the faces and the Interest packet must be deleted from the PIT. Thus, the PIT will be updated (decreasing), as described by Equation (4.13) and Equation (4.14), respectively.

$$PIT_{size} = PIT_{size} - p \tag{4.13}$$

$$VPIT_{size} = VPIT_{size} - p \tag{4.14}$$

The proceeding algorithms of AVPIT are presented at the receiving of both Data and Interest packets via Algorithms 4.1 and 4.2, respectively.

Algorithm 4.1 Treatment of Data packet in AVPIT

Input:
p : Entry
PIT_{size}: Number of packet currently in the PIT
$VPIT_{size}$: Number of packet currently in the VPIT

Main process:

```
Begin
 Check if Data packet name is matching any PIT entry name then
    Update the size of PIT based on PIT_size ← PIT_size − p // decreasing PIT size
    Update the size of VPIT based on VPIT_size ← VPIT_size − p // decreasing VPIT size
    Erase entry from PIT
    Cache the data in the Content Store
 end_if
   else
      Discard Data packet
    end_else
End
```

Algorithm 4.2 Treatment of Interest packet in AVPIT

Input:
p : Interest packet
s : VPIT arrival time
t : current time
C : Actual capacity
$\check{C}$: Virtual capacity
PIT_{size} : Number of packets currently in the PIT, initially equal 0
$VPIT_{size}$: Number of packets currently in the VPIT, initially equal 0
η_{PIT} : Limit size of PIT

Main process:
Begin
// at each interest packet arrival epoch do
Check **if** P_{CS}^{hit} equal zero **then**
Calculate the Interest arrival rate at PIT based on $\lambda_{PIT}^{in} \leftarrow \sum_{i\varepsilon|\eta_p} \lambda_i.(1-p_{CS}^{hit})$
end_if
Break
else
Calculate the process time based on t $\leftarrow h - s$
Update the size of the virtual PIT, calculating based on $VPIT_{size} \leftarrow VPIT_{size} - \check{C}.t$
Make the size of the virtual PIT has become zero, if virtual PIT size is less than zero
Check **if** the size of virtual PIT with incoming Interest packet size less than the limit size of PIT **then** // PIT does not overflow.
Add a new entry to the PIT // *normal NDN processing*
Update the size of PIT based on $PIT_{size} \leftarrow PIT_{size} + p$
Update the size of PIT based on $VPIT_{size} \leftarrow VPIT_{size} + p$
end_if
Go to step #
else
Check **if** the size of virtual PIT with new Interest packet size large than the size of PIT **then** // PIT does overflow
Make replacement entry in real PIT // *normal NDN process*
Return Interest NACK (PIT_Overflow) to downstream node
end_else
#: Recalculate the process time
Calculate the virtual capacity based on $\check{C} \leftarrow \check{C} + C.t$
Check **if** the virtual capacity large than actual capacity **then**
Make virtual capacity as the same actual capacity value
end_if
else
Update the virtual capacity based on $\check{C} \leftarrow \check{C} - p$
end_else
Make virtual capacity has become zero, **if** virtual capacity has less than zero
end_if
Update Interest packet arrive time by $s \leftarrow h$
end_else
End

4.2.3 AVPIT Verification and Validation

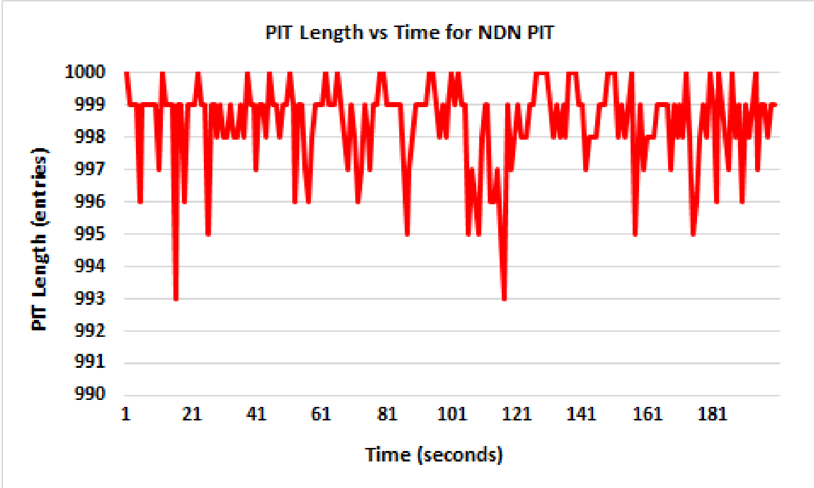
AVPIT mechanism was verified using the method described in Chapter Three and confirmed the following:

- AVPIT mechanism has been programmed correctly, and

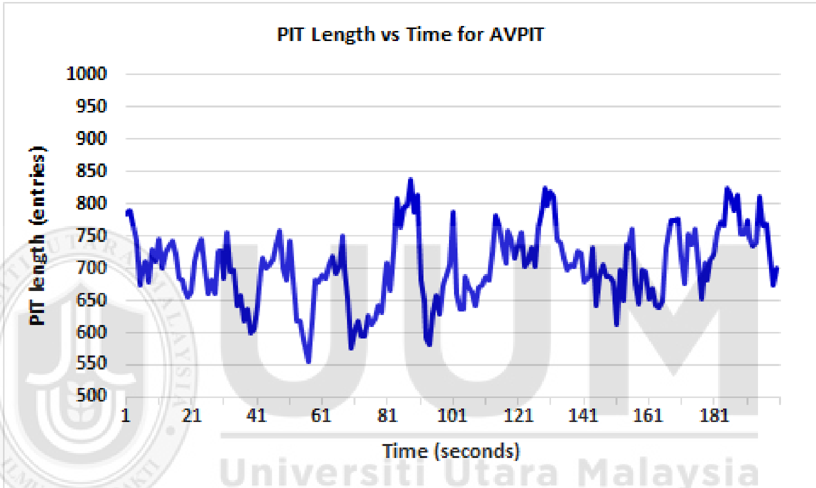
- AVPIT mechanism implementation is free from any errors or bugs.

To validate the accuracy of the proposed AVPIT mechanism, Graphical comparison technique is used. In this technique, the graphs of results that generated from simulating the model over time were compared with the graphs of results of valid mechanisms variables in order to investigate behaviors. In addition, a Dumbbell topology was considered with a variety of scenarios using the ndnSIM simulator. The former Dumbbell topology was used to study the impact of competing Interest packet rate on links and overflows of the PIT for 200 seconds of simulation time. A simple Dumbbell topology (see Figure 2.14) contains six nodes and five links, is selected.

Throughout this section, flooding route was used as the routing protocol in this scenario. Meanwhile, the maximum traffic load of 10000 Interest packet per second was set for all subscribers, and the PIT size is assumed to be 1000 entries. For this simulation scenario, the results obtained from AVPIT were compared with results obtained from traditional NDN PIT. This comparison was based on the behavior of an AVPIT in a certain heavy load (i.e., high Interest packet rate) for a particular performance metric. In other words, the AVPIT mechanism is expected to show a different behavior compared with the behavior of the traditional NDN PIT.



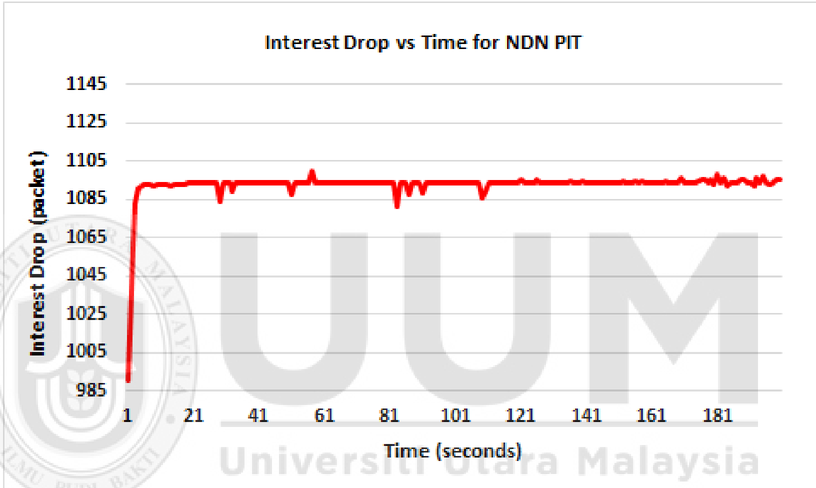
(a) PIT Length vs Time for NDN PIT



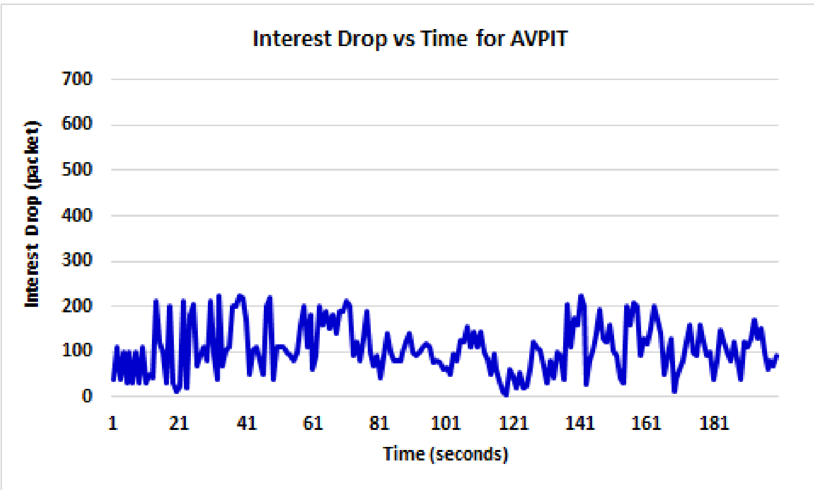
(b) PIT Length vs Time for AVPIT

Figure 4.3. PIT Length vs Time for NDN PIT and AVPIT

Figure 4.3 compares the validation of the real number of Interest packet caching in the PIT over time, both using and without using the AVPIT mechanism for NDN router based PIT overflow. In the Figure 4.3 (b), the fluctuations in the AVPIT's graph were not identical with the traditional NDN PIT graph. It can be observed that the PIT sized traditional NDN PIT fluctuates at a level that is close to the maximum capacity for the PIT, based on the simulation time when the PIT is overflowing. This is because the PIT controller needs only to replace the incoming new Interest packets according to the PIT replacement policy. On the other hand, by applying AVPIT, it can be observed that the PIT size fluctuates significantly up and down during the simulation time, due to the adaptive ability of the AVPIT based on the traffic load of the network and the VPIT conditions. When the VPIT becomes overflowed, the new Interest packet is discarded in the VPIT and the real Interest packet is replaced in the real PIT, while at the same time the NACK packet is sent to downstream nodes which maintains another rate limit accordingly. Hence, the figures above show that the overall behavior of AVPIT corresponds with the description and analysis of AVPIT, which supports that the AVPIT mechanism is valid.



(a) Interest Drop vs Time for NDN PIT



(b) Interest Drop vs Time for AVPIT

Figure 4.4. Interest Drop vs Time for NDN PIT and AVPIT

As seen in Figure 4.4 after an initial transient period, it can be observed that the NDN PIT Interest packet drop is completed with the Interest drop range between (1072 and 1089) Interest packets, and it fluctuates in this range. Hence, the fluctuation of the Interest packet drop is due to the packet nature of the NDN PIT processing for the arrival of Interest packets when the PIT suffers overflow. On the other hand, it can be observed that the AVPIT (see Figure 4.4 (b)) Interest drop is completed with the Interest drop range between (6 and 237) Interest packets, while fluctuating in this range. By reducing the Interest drop that may occur over heavy traffic load, significant cost can be reduced in both packet processing overhead and storage in the NDN router. Hence, it is proven here that AVPIT prevents unnecessary traffic in the network.

4.2.4 AVPIT Evaluation

With the aim of evaluating the performance aspects of the AVPIT mechanism, a comparison using ndnSIM was conducted using the traditional PIT in the NDN router. For this purpose, a Tree topology (see Figure 2.15). For each simulation scenario, the experiment was repeated 10 times for 300 seconds of simulation time. These simulations were performed using the NDN nodes. In the presented scenario, there were two different performance metrics, namely the real amount of Interest caching in PIT and Interest drop at the PIT, which were analyzed to investigate the performance of the AVPIT.

Figure 4.5 shows the real number of Interest caching in PIT using and without using the AVPIT mechanism for the NDN router (router R) over a Tree topology. The horizontal x-axis represents the different PIT sizes (i.e., 100, 200,300, 400, 800, 1000, 1200, 1600, and 2000 entries), while the vertical y-axis represents the real number of Interest packet caching in PIT. Meanwhile, the other parameters were selected as CS size = 1000 entries, and the Interest packet rate = 10000 Interest/second. As far as traditional NDN PIT is concerned, the real number is almost close to the capacity of PIT when setting a small PIT size scenario. In contrast, the AVPIT average PIT size would increase gradually from approximately (85 entries) to around (1541 entries). Interestingly, the AVPIT greater reduces the real number of Interest packet caching in PIT as compared to traditional NDN PIT.

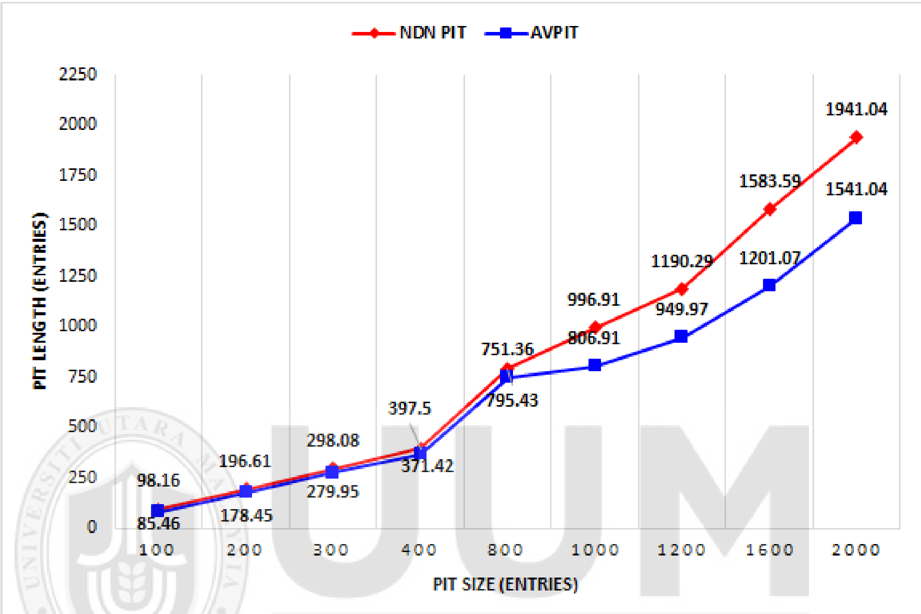


Figure 4.5. PIT Length vs PIT Size for NDN PIT and AVPIT

Figure 4.6 illustrates the Interest drop rate differences of AVPIT and traditional NDN PIT over 10 different Interest packet rates (i.e., 1000-10000 Interests/second), whereas the other parameters were selected as PIT size = 1000 entries, and CS = 1000 entries. The vertical y-axis represents the Interest drop rate, while the horizontal x-axis represents the different Interest packet rate per second. Both groups showed a gradual rise in Interest drop as the Interest packet rate increased, and as far as traditional NDN PIT is concerned, the Interest drop fluctuated between (22.74%) and (47.80%). In contrast, the AVPIT Interest drop increased gradually from approximately (14.25%) to around (29.98%). Thus, it was observed that the Interest drop for both mechanisms linearly increased with the increase in Interest packet rate by the subscribers. Interestingly, the

AVPIT greater reduces the Interest drop rate as compared to traditional NDN PIT.

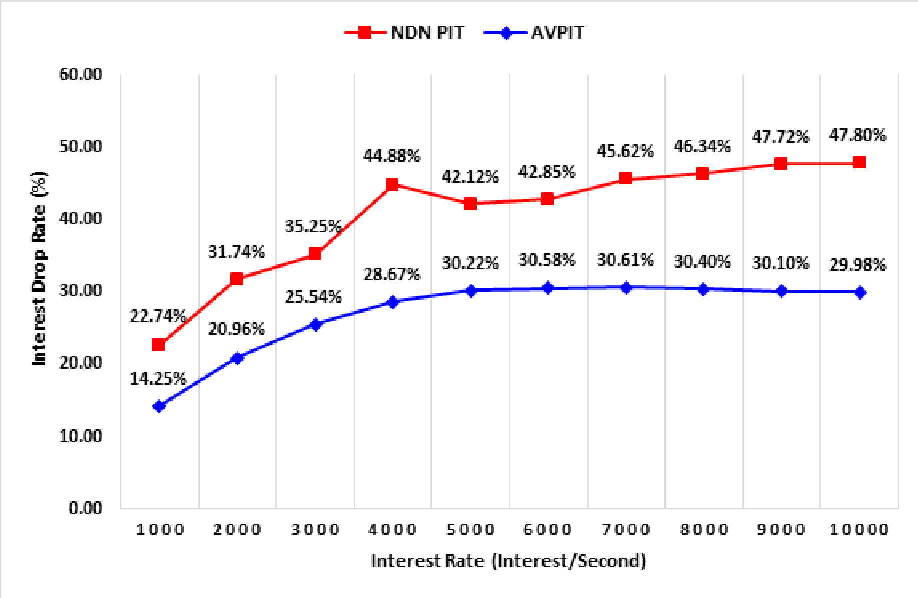


Figure 4.6. IDR vs No. of Interest Rate for NDN PIT and AVPIT

As a conclusion, the traditional NDN PIT and AVPIT were compared using a Tree topology and the results are presented in Figures 4.5 and 4.6. In particular, it is claimed here that AVPIT may be used in order to reduce the PIT overflow, and is able to maintain a real number of Interest packet caching in PIT with minimal Interest packet drop at the PIT, when compared to traditional NDN PIT.

4.3 Smart Threshold Interest Lifetime Mechanism

Implementing an PIT needs large and fast memory configuration to store the incoming Interest packets at a particular point in time. In addition, the entry is lodged in the PIT for a time interval referred to as lifetime (i.e., Interest lifetime value is indicated as the approximate time remaining before the Interest packet times out); this is specified as an inbuilt parameter of the Interest packet. As soon as the Interest packet lifetime expires, while the response still remains pending, the holding memory is thus freed to accommodate an incoming Interest packet. Interestingly, it has been noted that the lifetime parameter is usually chosen by the subscribers. Moreover, the PIT size could amount to a bottleneck problem for the NDN infrastructure when large sets of requests flood the network and Data packets are returned with a delay.

Furthermore, it is predicted that the use of longer Interest packet lifetime which would further increase the number of simultaneous entries in the PIT. Despite the diversified nature of the NDN router, sometimes a large lifetime is still essential since subscriber nodes would subscribe for some given contents that are asynchronously produced in future requests. In this section, this specific problem motivated the researcher to provide a Smart Threshold Interest Lifetime (STIL) mechanism, which defines the management of an Interest packet lifetime at content PIT. The STIL can change the Interest packet lifetime value (granted to be larger) during low network load, and thus the PIT is not overflowed. In contrast, STIL can change Interest packet lifetime value (granted to be less) during heavy network load, and the PIT is overflowed without increasing Interest packet retransmission. This is in order to avoid overstaying of PIT entries. The significance of this section is that it presents the STIL model, the theory, analysis of STIL, and validation and evaluation of the proposed mechanism.

4.3.1 Description of STIL Mechanism

The initial design of the PIT memory size was to determine the number of entries and the lifetime of each entry [173]. Thus, the adaptation of Interest packet lifetime is helpful to utilize the PIT memory size sufficiently. According to Anastasiades *et al.* in [69] in the NDN nodes' prototype implementation (i.e., CCNx [302]), the default lifetime for Interest packet is 4 seconds (i.e., 4000 ms). However, this parameter is chosen by subscribers, and not by the network controller itself. The problem can exacerbate via long Interest packet lifetime . In order to deal with the aforementioned problem, a need arose to develop a special mechanism to manage Interest packet lifetime (i.e., STIL) at content router levels by adapting Interest packet lifetime as a function under the NDN network load. Hence, STIL can adjust the Interest packet lifetime in order to avoid overstaying of PIT entries during heavy network load, which is the main goal for this mechanism.

Based on the discussion above, the STIL mechanism breaks down the hypothesis that the intermediate nodes (NDN routers) do not manage Interest packet lifetime values. Therefore, STIL mechanism answer of the second question of the research questions, which it is highlighted the direct effect of massive use of long Interest lifetime on the PIT size, that leads to increase the probability this table become overflowed. The main goals of STIL are:

- to observe the incoming faces list in order to determine which face has a highest receiving Interest packet in the NDN router (if and only if there are variety number of Interest packet received via the NDN router faces), and assist in determining the Interest packet that is coming from the face, and
- to manage the incoming Interest packet lifetime to PIT, as a function of the network load.

4.3.2 Analysis of STIL Mechanism

The Renewal theory behind the STIL mechanism addresses issues of both Interest packet lifetime and PIT size. In addition, the update procedure depends on the incoming Interest packets, which implies that every time packets arrive to the PIT, an update is required. The former issue can be resolved by assigning a specific parameter, while the latter is settled by an estimation using Renewal theory and storing the entry based on the value of Interest packet lifetime and face behavior. An adaptive mechanism for the Interest packet lifetime is beneficial in NDN router scenario to control how quickly retransmission can be performed in case of PIT overflow. This is considered as the beginning of the analysis of the newly developed mechanism (STIL) to adaptively set the lifetime for incoming Interest packets. Moreover, in order to support this adaptive mechanism, the face that has a higher number of Interest packet received needs to be determined. It is assumed that f_{max} is the face number that has received a maximum Interest packet for set (R). Whereas R is a set of ratios of the number of requests received from a given face (η_r) to the total number of requests received from all faces (η_R), which is calculated as follows:

$$f_r = \left(\frac{\eta_r}{\eta_R}\right) \tag{4.15}$$

where f_r is the requests ratio via face f that is belonging to the set R.

$$f_{max} = max(f_{ri}) \tag{4.16}$$

where f_{ri}=$\{f_{ri} | f_{ri} \subseteq R, \forall i \varepsilon \eta_{face}\}$. When $f_{max} > 0$, the function returns either the face number which has received a maximum number of Interest packet or 0 that represents all faces have received the same number of Interest packet.

As an Interest packet arrives, three cases based on the relationship between the PIT and Interest packet lifetime can occur.

- ***Case 1:*** the PIT is empty, let (T_{th}) be denoted as a threshold of an average lifetime that represents an entry lifetime between the T_{th} values, which initially is assumed as a default set to 4000 ms with the first incoming Interest packet lifetime to the PIT given as:

$$T_{th} = \frac{4000 + p_\tau}{2} \tag{4.17}$$

where p_τ is the lifetime of an incoming Interest packet.

- ***Case*** 2: the PIT is not overflowed and the PIT is not empty, so the T_{th} is updated by calculating the average lifetime between the T_{th} values, with the incoming Interest packet lifetime to the PIT as follows:

$$T_{th} = \frac{T_{th} + p_\tau}{2} \tag{4.18}$$

Hence, the value of lifetime of an incoming entry is updated as:

$$p_\tau = max(T_{th}, p_\tau) \tag{4.19}$$

Since T_{th} is larger than the incoming Interest packet lifetime, then p_τ will be updated based on (4.19). Hence, by applying the Equations (4.18) and (4.19) for all incoming Interest packets to the PIT, it motivates the entry to keep alive as long as possible before the expiration of entry lifetime.

- ***Case 3:*** the PIT is overflowed, thus the average lifetime for all entries that belong to face (f) is calculated (when the number of Interest packet received via faces is varying) or for all entries of PIT (when the number of Interest packet received via faces is similar), it is subtracted based on the equation from the old threshold, which is as follows:

Theorem 4.1: The threshold lifetime (T_{th}) for all entries belonging to PIT is given by:

$$T_{th} = \begin{cases} T_{th}(f_{max}) = \frac{\sum_{i\varepsilon|w} p_{\tau i}}{w} & if \; f_{max} > 0 \\ T_{th}(PIT) = \frac{\sum_{i\varepsilon|PIT_{size}} p_{\tau i}}{PIT_{size}} & Otherwise \end{cases} \tag{4.20}$$

where w is the total number of entries $\varepsilon(f_{max})$; and PIT_{size} is the total number of entries $\varepsilon(PIT)$. Thus, the STIL updates the incoming PIT entry lifetime based on the Interest packet received via the face and PIT conditions.

Proof: Theorem 4.1 shows a rule of updating the new incoming PIT entry lifetime based on updating the T_{th} parameter of either the specific given face (i.e., f_{max}), or for all entries belonging to the PIT. Thus, the STIL algorithm is implemented for monitoring and managing PIT for each incoming PIT entry lifetime when the PIT becomes overflowed. Therefore, the STIL will make a choice of selecting from the two available options mentioned above. When there are different number of Interest packet received via the faces, Equation (4.20) is determined, and as mentioned above, the f_{max} returns the face number at the maximum number of Interest packet arrival. Let $T_{th}(f_{max})$ denote the average lifetime that the PIT receives from the face (f_{max}), that is still stored in the PIT. Therefore, $T_{th}(f_{max})$ is defined by:

$$T_{th}(f_{max}) = \frac{\sum_{i\varepsilon|w} p_{\tau i}}{w} \tag{4.21}$$

where $i = \{1, 2, 3, ..., w\}$, and w is the total number of entries $\varepsilon\,(f_{max})$ inside the PIT. When the number of Interest packet received is similar over all faces, $T_{th}(PIT))$ denotes the average lifetime for all entries into the PIT. Therefore, $T_{th}(PIT)$ is defined by:

$$T_{th}(PIT) = \frac{\sum_{i\varepsilon|PIT_{size}} p_{\tau i}}{PIT_{size}} \tag{4.22}$$

where $i = \{1, 2, 3, ..., PIT_{size}\}$ and PIT_{size} is the total number of entries $\varepsilon\,(PIT)$. Consequently, either one of these equations is true (i.e., (4.21) or (4.22) are mutually exclusive). Accordingly, the expected value of the calculated lifetime is as follows:

$$T_{th} = \left[\frac{\sum_{i\varepsilon|w|PIT_{size}} p_{\tau i}}{w|PIT_{size}}\right]^{f_{max}|PIT} \tag{4.23}$$

$[x]^{f_{max}|PIT}$ represents the update of the lifetime of the newly incoming entry for face (f_{max}) or PIT. During overflow, the value of lifetime of an incoming Interest packet is updated based on Equation (4.24) as follows:

$$p_\tau = min(T_{th},\, p_\tau) \tag{4.24}$$

This can be helpful by granting shorter lifetime values in case of high traffic loads, and when the PIT becomes full. Hence, by applying Equation (4.23), it motivates the entry to keep alive for the shortest acceptable period before the expiration of entry lifetime. Initially, STIL was implemented and developed based on traditional NDN PIT. Thus, STIL will be activated when the real PIT becomes overflowed. This investigation further proceeds with the algorithm of STIL presented in Algorithm 4.3:

Algorithm 4.3 Treatment of entry lifetime in STIL

Input:
T_{th} : Threshold lifetime of PIT, initially equal 4000 Or based on subscribers setting
p_τ: Entry lifetime
p: Incoming entry

Main process:
Begin
if the PIT does not overflow **then**
    if the PIT is empty **then**
    Update by T_{th} calculating the average between the *4000* and incoming Interest lifetime
    end_if
    else *// PIT not empty*
    Update T_{th} by calculating the average between the T_{th} and incoming Interest lifetime based on $T_{th} = \frac{T_{th}+p_\tau}{2}$
    Update the entry lifetime based on $p_\tau \leftarrow max(T_{th}, p_\tau)$
    end_else
end_if
Break
else // PIT overflow
    Check **if** there is any face has received highest number of Interest arrival **And** the incoming Interest belong to this face **then**
    for all entities belong the face has received highest number of Interest packet into PIT **do**
    Calculate the threshold lifetime parameter based on $T_{th} \leftarrow \frac{\sum_{i\varepsilon|w} p_{\tau i}}{w}$
    end_for
    end_if
    Go to step *
    else // The number of Interest packet received is the same for all faces **Or** the incoming Interest not belong to this face
    for all entities belong in PIT **do**
    Calculate the threshold lifetime parameter based on $T_{th} \leftarrow \frac{\sum_{i\varepsilon|PIT_{size}} p_{\tau i}}{PIT_{size}}$
    end_for
    end_else
*: Update the entry lifetime based on $p_\tau \leftarrow min(T_{th}\, p_\tau)$
end_else
End

4.3.3 STIL Verification and Validation

The primary concern is to ensure that the proposed STIL has been programmed correctly inside the ndnSIM, and it has been implemented properly on the computer. Verification was conducted following the techniques mentioned in Chapter Three and aimed at confirmed the following:

• STIL mechanism is programmed correctly, and

• STIL mechanism implementation does not contain any errors or bugs.

It was verified that PIT overflow leads to decreased PIT utilization. Therefore, this section focuses on an adaptive lifetime parameter in order to avoid Interest packet stay for a long time inside PIT using the proposed STIL mechanism. The validation of STIL was intended to ensure that it meets its intended requirements in terms of adapting the lifetime of Interest packet based on the status of the PIT. Thus, the validation of STIL focused on examining the relationship between PIT overflow and the lifetime value over two different ways, including a numerical analysis, and simulation scenarios.

Numerical analysis is a method that is used to validate the analytical model by assuming valid input and calculating them based on the model equations. These methods are repeated 10 times with different inputs to ensure the model meets the intended requirements in terms of adapting the Interest packet lifetime based on the PIT condition (in worst case; case 3, when the PIT is full). The limited PIT was size=10 entries, and the replacement policy was LRU. For NDN router (R), the number of faces were assumed to be three at which $f = \{1,2,3\}$ (i.e., router A: face 1, router B: face 2, and router C: face 3). In addition, the Interest packets were set (S) to 20, as follows: S={Interest lifetime, incoming face}={{3240, 3}, {3087, 3}, {3310, 2}, {3040, 3}, {3101, 1}, {3178,3}, {3215, 3}, {3371, 1}, {3214, 3}, {3222, 2}, {3190, 3}, {3195, 3}, {3063, 2}, {3249, 3}, {3314, 1}, {3388, 3}, {3053, 2}, {3211, 3}, {3107, 2}, {3195, 3}}.

As presented earlier in Section 4.3.3 above, the case 3 of STIL mechanism involves two options. In option 1 for case 3 (not on the order as explained in the equation), it was assumed that the ratios of the number of requests received in the list of incoming faces are similar ($A = B = C$). This means that the face field from the data set will be ignored and the focus is only on the lifetime field. Table 4.1 illustrates the results that reflect STIL mechanism process.

Table 4.1
PIT Process After Overflow based on STIL Mechanism

	The Lifetime of Incoming Interest Packets (ms)									
Incoming entry	3190	3195	3063	3249	3314	3088	3053	3211	3107	3195
PIT Entries	3240	3190	3190	3190	3190	3190	3190	3190	3190	3190
	3087	3087	3193	3192	3192	3192	3192	3192	3192	3192
	3310	3310	3310	3063	3063	3063	3063	3063	3063	3063
	3040	3040	3040	3040	3179	3178	3178	3178	3178	3178
	3101	3101	3101	3101	3101	3193	3192	3192	3192	3192
	3178	3178	3178	3178	3178	3178	3088	3088	3088	3088
	3215	3215	3215	3215	3215	3215	3215	3053	3053	3053
	3371	3371	3371	3371	3371	3371	3371	3371	3176	3176
	3214	3214	3214	3214	3214	3214	3214	3214	3214	3107
	3222	3222	3222	3222	3222	3222	3222	3222	3222	3222
Avg. Lifetime	3198	3193	3203	3179	3193	3202	3193	3176	3157	3146
Threshold	3198	3193	3203	3179	3193	3202	3193	3176	3153	3146

In order to detail how the mechanism works, an example of one operation is described in the following. The PIT keeps receiving the Interest packet, and updates are computed until the PIT becomes full. This means that the PIT table includes 10 entries. Next, the newly incoming entry is recorded (p_{τ}= 3190 ms). STIL mechanism calculates the average entry lifetime, which results in 3198 ms. Then, it updates T_{th} in order to allow only the entry that has lifetime less than the T_{th} to replace the old one. Hence, in this scenario as an example, there is no need to update the lifetime field at the initial iteration. Otherwise, the incoming entry lifetime is updated with the T_{th}.

In addition, in order to juxtapose that it fulfills its intended requirements in terms of adaptive lifetime, the STIL mechanism results were evaluated with the original PIT (i.e., PIT without STIL mechanism, as shown in Table 4.2).

Table 4.2

PIT Process After Overflow based on NDN PIT

	The Lifetime of Interest Incoming Packets									
Incoming entry	3190	3195	3063	3249	3314	3088	3053	3211	3107	3195
PIT Entries	3240	3190	3190	3190	3190	3190	3190	3190	3190	3190
	3087	3087	3195	3195	3195	3195	3195	3195	3195	3195
	3310	3310	3310	3063	3063	3063	3063	3063	3063	3063
	3040	3040	3040	3040	3249	3249	3249	3249	3249	3249
	3101	3101	3101	3101	3101	3314	3314	3314	3314	3314
	3178	3178	3178	3178	3178	3178	3088	3088	3088	3088
	3215	3215	3215	3215	3215	3215	3215	3053	3053	3053
	3371	3371	3371	3371	3371	3371	3371	3371	3211	3211
	3214	3214	3214	3214	3214	3214	3214	3214	3214	3107
	3222	3222	3222	3222	3222	3222	3222	3222	3222	3222
Avg. Lifetime	3198	3193	3204	3179	3200	3221	3212	3196	3180	3169

Figure 4.7 reflects the results obtained in Table 4.1 and Table 4.2. The horizontal an x-axis represents the PIT update process per cycle, while the vertical an y-axis represents the average entry lifetime. which shows that the average entry lifetime for PIT with STIL mechanism resulted in a better outcome. The scenario here supposed that all 10 processes were controlled, lifetime was updated, and there was a replacement for each

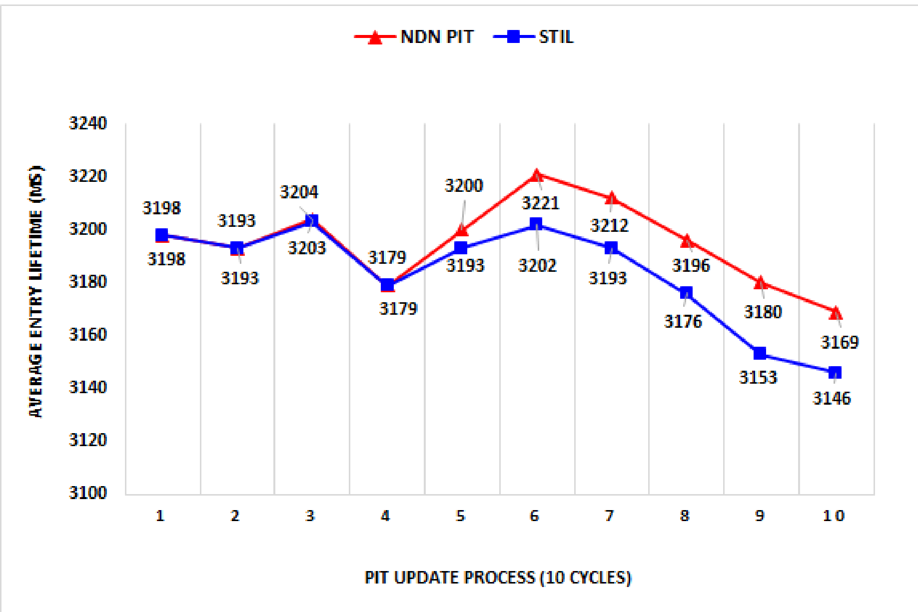


Figure 4.7. Average Entry Lifetime vs PIT Update Process for NDN PIT and STIL with Similar Interest Rate

incoming Interest packet when the PIT overflows. Thus, the lifetime parameter is decreased for all incoming Interest packets that were replaced by the existing one.

In option 2 for case 3: it is assumed that the ratios of the number of requests received in the list of incoming faces are varied (for instance $A = \frac{3}{20}$, $B = \frac{5}{20}$, $C = \frac{12}{20}$, see data set S), which implies that the focus is on both the lifetime field and the face field as well. For illustration purposes, Table 4.3 depicts the result that conflicts with the mechanism process by applying the same concept as in case one, until the PIT becomes full.

Table 4.3
PIT Process After Overflow based on STIL Mechanism (detected face)

The Lifetime of Incoming Interest Packets (ms)																				
Incoming entry	3190	*f*	3195	*f*	3063	*f*	3249	*f*	3314	*f*	3366	*f*	3053	*f*	3211	*f*	3107	*f*	3195	*f*
PIT Entries	3240	3	3162	3	3163	3	3163	3	3163	3	3163	3	3163	3	3163	3	3163	3	3163	3
	3087	3	3087	3	3150	3	3150	3	3150	3	3150	3	3150	3	3150	3	3150	3	3150	3
	3310	2	3310	2	3310	2	3063	2	3063	2	3063	2	3063	2	3063	2	3063	2	3063	2
	3040	3	3040	3	3040	3	3040	3	3160	3	3160	3	3160	3	3160	3	3160	3	3160	3
	3101	1	3101	1	3101	1	3101	1	3101	1	3184	1	3184	1	3184	1	3184	1	3184	1
	3178	3	3178	3	3178	3	3178	3	3178	3	3178	3	3180	3	3180	3	3180	3	3180	3
	3215	3	3215	3	3215	3	3215	3	3215	3	3215	3	3215	3	3053	2	3053	2	3053	2
	3371	1	3371	1	3371	1	3371	1	3371	1	3371	1	3371	1	3371	1	3173	3	3173	3
	3214	3	3214	3	3214	3	3214	3	3214	3	3214	3	3214	3	3214	3	3214	3	3107	2
	3222	2	3222	2	3222	2	3222	2	3222	2	3222	2	3222	2	3222	2	3222	2	3222	2
Avg. Lifetime	3198		3190		3196		3172		3184		3192		3192		3176		3155		3146	
Threshold	3162		3150		3196		3160		3184		3180		3192		3173		3155		3173	

The treatment of this option determines the face that has the highest ratio that the ratios of the number of requests received in the list of incoming faces. STIL mechanism calculates the T_{th} for every entry recorded inside the PIT, which in this case was equal to 3162 ms. Then, T_{th} is updated in order to allow only the entries that have lifetime values less than the T_{th} to be replaced with old ones based on the LRU policy. Conclusively, in this example T_{th} = 3162 ms, which means it requires an update of the p_t for newly requested Interest packets and passes it to the PIT.

In addition, as illustrated in Figure 4.8, the performance of PIT with STIL mechanism has a good average entry lifetime. This is because the incoming Interest packet lifetime is updated based on the T_{th}, by calculating the average for all packets that belong to the same face with a maximum value. Thus overall, the lifetime parameter is decreased for all incoming Interest packets that are replaced by an existing one.

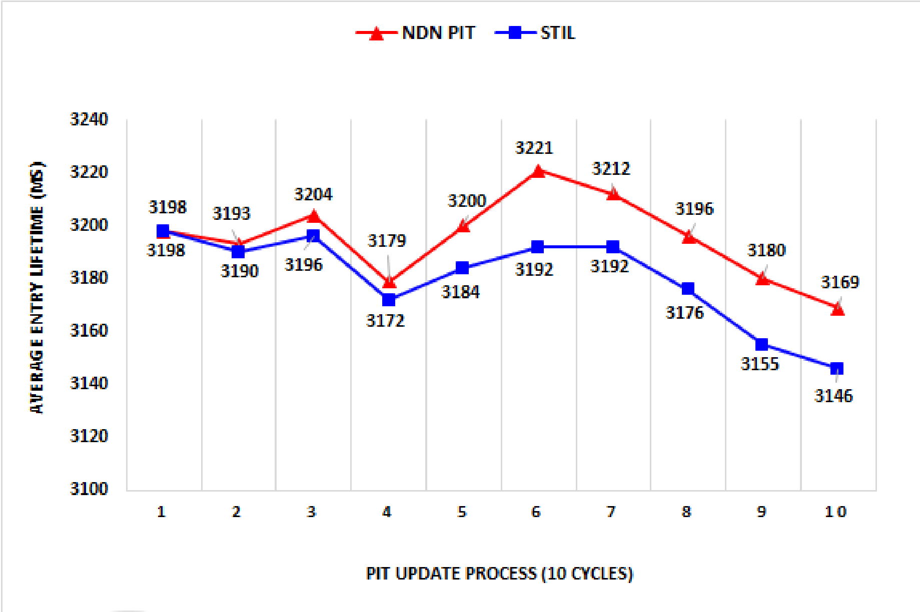
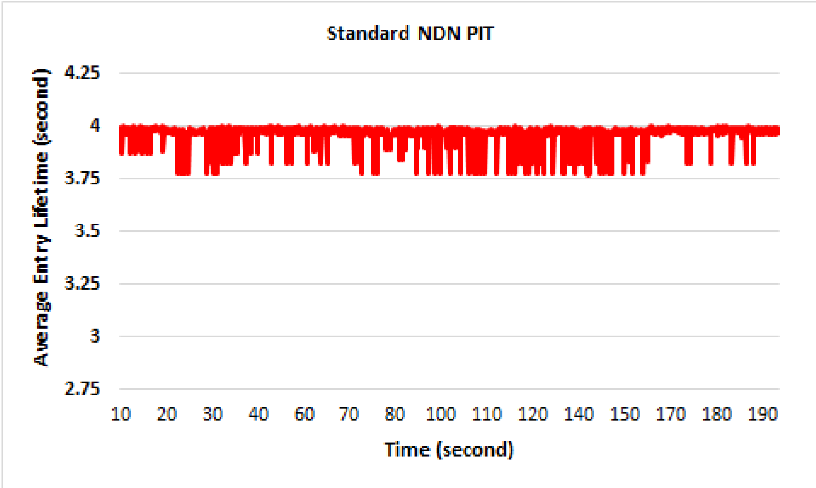
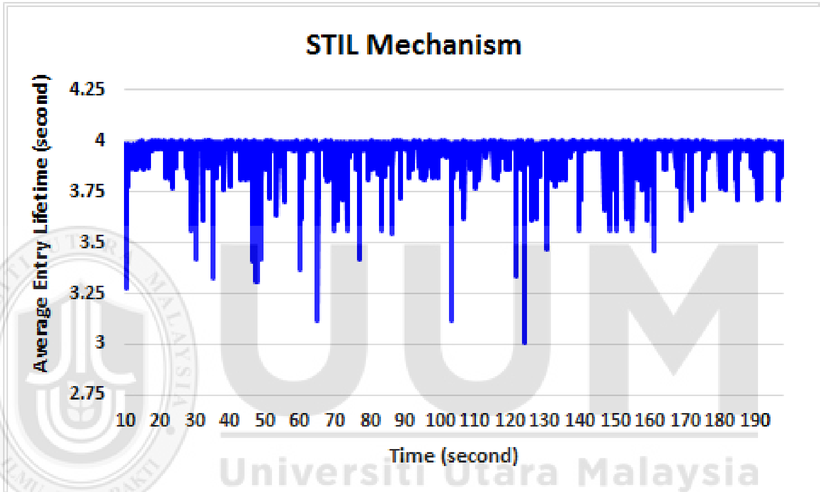


Figure 4.8. Average Entry Lifetime vs PIT Update Process for NDN PIT and STIL with Various Interest Rate

Meanwhile, the simulation scenario was used as another way of validating the STIL mechanism based on Performance testing technique, which is used to test whether all characteristics of the performance are measured as well as evaluated with enough accuracy. For that, we are comparing the results obtained from STIL with the results from standard NDN PIT via Tree topology. Flooding route was used as the routing protocol in this scenario, while the PIT size was set at 100 entries, and the CS at 100. In addition, the Interest packet rate was set at 1000 Interest/second as the system parameter. Interest packet lifetime denotes the initial contention at 4 seconds (i.e., 4000 ms). For each simulation scenario, the result from the mechanism model was compared with traditional NDN PIT results. For each simulation scenario, the experiment was repeated ten times as well as the simulation time set at 200 second.



(a) Average Entry Lifetime vs Time for NDN PIT



(b) Average Entry Lifetime vs Time for STIL

Figure 4.9. Average Entry Lifetime vs Time for NDN PIT and STIL

Figure 4.9 represents the average entry lifetime of the NDN PIT and STIL versus simulation time. Thus, NDN router (R) maintains the lifetime of the incoming Interest packet based on network behavior and PIT condition as described in Section 4.3.3. The figure shows that the STIL maintains the Interest lifetime at maximum as long as possible since the PIT has not overflowed yet. STIL adapts the lifetime of incoming Interest packets when the PIT overflows as the minimum Interest packet lifetime of (3.00999) seconds as compared to (3.99999) seconds for NDN PIT. It was also observed that the incoming Interest packets was decreased by approximately (31%) as compared to the results obtained using NDN PIT.

4.3.4 STIL Evaluation

To assess the performance of the proposed STIL mechanism, extensive NDN simulations were carried out in different network scenarios using ndnSIM simulations. In this section, a selected set of simulation results is presented in illustrating the function, properties, and benefits of the proposed solution. The experimental evaluation investigates the impact of STIL on the performance of PIT by comparing the STIL results with traditional NDN PIT and DPEL [61] output. In spite of the simplified implementation and simulation of DPEL mechanism, it has not been compared and evaluated in wire environments. Consequently, DPEL was implemented in the present study in ndnSIM and it was evaluated by selected simulation scenarios. The PIT size and CS size were set as 100 Interest packets and 100 Data packets, respectively.

For the purpose evaluation of STIL, Interest packet lifetime is assuming to be 80 ms [52] rather than 4000 ms in order to be sure that STIL not increase the Interest retransmission when the PIT become full. The Interest packet rate was set as 250, 500, 750, 1000, 1250, 1500, 1750, and 2000 Interests/second. In order to model a more realistic network environment, the Abilene topology was implemented, as shown in Figure 2.16. The number of subscriber nodes and publisher nodes were fixed to 25 and 10, respectively. Each subscriber/publisher was connected to one NDN router uniformly selected between those available in the network. For each simulation scenario, the experiment was repeated ten times as well as the simulation time set at 500 second. Moreover, two different performance metrics, which were average Interest packet retransmission rate, and the Interest packet satisfaction rate were analyzed to study the performance of STIL.

Figure 4.10 shows the Interest retransmission rate of STIL, DPEL, and NDN PIT over different subscriber Interest packet rates. The horizontal an x-axis represents the eight different Interest packet rates per second, while the vertical an y-axis represents the Interest retransmission rate. Subscribers resent Interest packets that have been damaged and lost. Meanwhile, Interest packet retransmission after a time-out is necessary to recover losses, even though the Data packet may just be delayed at the bottleneck and/or the Interest packet is dropped when the PIT capacity is not sufficient. Thus, subscribers' Interest packets will be discarded from the NDN routers, and based on this, subscribers will experience an increasing Interest packet retransmission that will result in a complete collapse of the whole network. According to the results obtained, STIL, DPEL, and NDN PIT share the same trend with the Interest packet retransmission linearly increasing with the increase in the number of Interest packet rate provided by the subscriber. The graph below illustrates that when STIL is applied, the drop packets are minimal, and the retransmissions required is at the minimum, thus saving the Interest packet retransmission costs. Hence, the STIL mechanism presented the best results as compared with DPEL and NDN PIT.

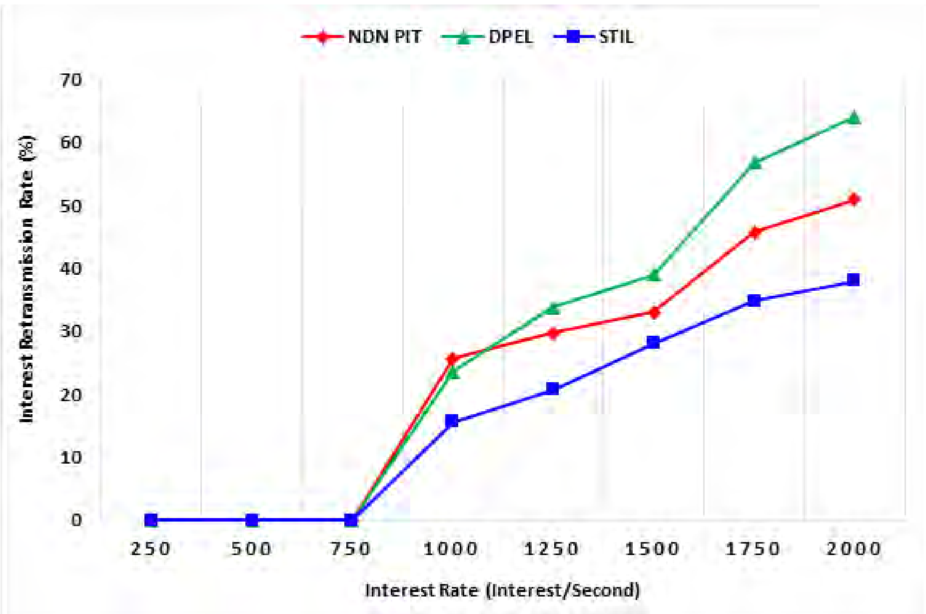


Figure 4.10. IRR vs Interest Rate for NDN PIT, DPEL and STIL

Figure 4.11 illustrates the Interest packet satisfaction rate of STIL, DPEL, and NDN PIT over the Abilene topology. The horizontal an x-axis represents the eight different Interest packet rates per second, while the vertical an y-axis represents the Interest packet satisfaction rate. In scenarios where the Interest rates were between 250-2000 Interest/second, different Interest packet satisfaction rate values were set for different predictors. The Interest packet satisfaction for STIL was observed to be always better than NDN PIT and DPEL. This is because the STIL mechanism has the ability of adaptive Interest packet lifetime setting in both cases; when the PIT does/does not overflow. Hence, this guarantees that with a large heavy load, the Interest packet satisfaction can be dramatically maximized as compared to NDN PIT and DPEL, respectively.

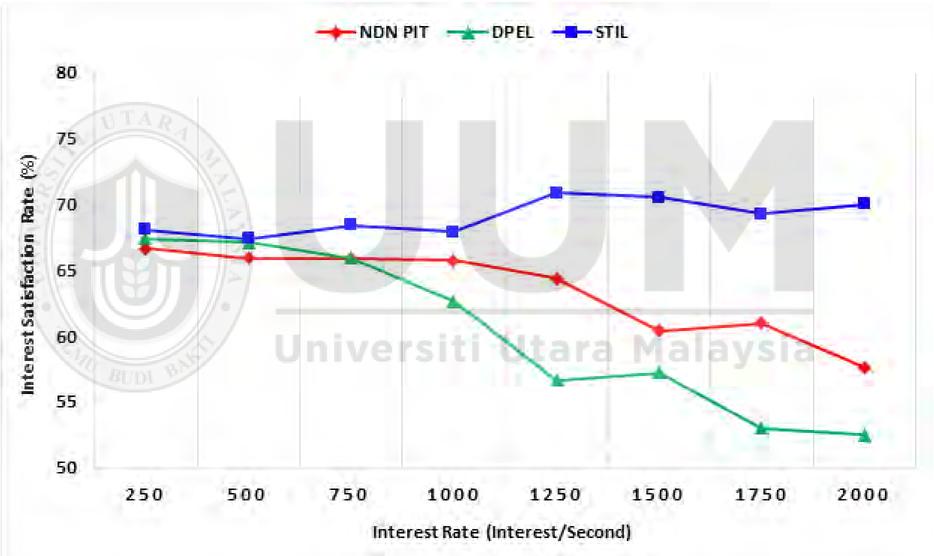


Figure 4.11. ISR vs Interest Rate for NDN PIT, DPEL and STIL

As a conclusion, STIL, DPEL and NDN PIT were compared by using an Abilene topology and the results are shown in Figures 4.10 and 4.11. The analysis shows that STIL achieves a higher Interest satisfaction when compared with NDN PIT and DPEL, respectively. In addition, STIL significantly reduces the Interest retransmission rate that is experienced by the subscriber node(s) when the Interest packet rate increases.

4.4 Highest Lifetime Least Request Policy

The PIT replacement policy plays a crucial part in Interest packet caching. To achieve a high level of sophisticated cache management, these replacement policies are necessary. Due to the limited size of the PIT, PIT replacement policies are helpful in evicting the entry from the PIT and building a new space for the new Interest packet coming to the PIT. This is applicable when the PIT is full and then there is an incoming entry that needs to be inserted into the PIT. In the quest for obtaining a replacement policy that is as close to efficient as possible, this section presents a new replacement policy, namely Highest Lifetime Least Request (HLLR) policy. HLLR is a replacement policy that determines a PIT entry to be changed based on PIT conditions for efficient PIT utilization. The significance of this section is that it presents the HLLR model, the theory, analysis of HLLR, and validation and evaluation of the proposed policy.

4.4.1 Description of HLLR Policy

In the NDN architecture, the management of NDN routers' storage capacity plays a fundamental role in system performance [92]. Thus, management of PIT continues to be one of the primary concerns of high-speed forwarding. For content PIT replacement, different cache replacement policies have been proposed, which are Persistent, Random, and Least Recently Used. For Persistent policy, a new entry will not be created if a limit has been reached. Regarding Random policy, when PIT reaches its limit, a random entry (both old and new entries) could be removed from the PIT. As for Least Recently Used, when limit is reached, insertion of a new entry will evict the oldest entry [218, 47]. However, without consideration of other factors, such as being accessed frequency, it is unable to get an ideal effect in some modes. In addition, the lifetime parameter was not considered previously, even though it can affect entries with long entry lifetime. Therefore, it deserves a specific study, which is the object of this thesis.

To overcome this sort of problem, a Highest Lifetime Least Request (HLLR) policy was proposed in order for efficient utilization of PIT via determining which PIT entry needs to be replaced based on PIT conditions. Hence, this policy was proposed to answers the third question of the research questions, which it is highlighted the direct effect of Interest lifetime and popularity of PIT content in replacing PIT entry when the PIT is overflowed for efficient PIT utilization. The main goals of HLLR were:

- to penalizing the Interest packet that occupies a large Interest packet lifetime with a minimum number of request of Interest packet in the PIT, and
- to formulate a general rule that is a function of the entry lifetime and frequency of entry that yields an efficient priority for replacement.

4.4.2 Analysis of HLLR Policy

The HLLR policy is a type of memory management algorithm used to manage PIT within NDN routers. The characteristics of this algorithm involve the PIT that keeps track of the number of times an Interest packet is requested in memory by assigning a counter to count every face that requests the Interest packet from the PIT. Each time a reference is made to that Interest packet, the counter is increased by one. When the PIT is full and has a new Interest packet waiting to be inserted, the HLLR will search for the Interest packet with the least face frequency and highest lifetime among all the entries present in the PIT. The least face frequency and highest lifetime are used to determine the PIT entry to be changed.

As clarified above, HLLR has two vectors which are entry lifetime and the number of entry request. When the PIT overflows, one entry is replaced with a newly arrived one. HLLR is designed to determine the entry that has longest lifetime and least faces requesting to be replaced. Each entry is supposed to have a scalar entry weight of flowing measure, namely "*weight*" and denoted by$p(weight)$, where $p(weight)$ represents the weight of PIT entry that needs to be evicted (i.e., time to store one request for entry into PIT), which is defined as:

$$p(weight)_{PIT} = \frac{p_\tau}{r_c} \tag{4.25}$$

where p_τ presents the entry lifetime, and r_c is the frequency of the entry that is already cached in the PIT, which is calculated as:

$$p(r_c) = count\{c, p : 0 < c \leq \eta_{face}\} \tag{4.26}$$

where c is the request counted for the entry; c is greater than 0 and less or equal to maximum number of faces (η_{face}) of the NDN router. However, Equation (4.25) does not support the correct $p(weight)$ in some cases, particularly when there are two results having the same value (i.e., $p(weight) = \frac{p_\tau}{r_c} = \frac{150}{2} = \frac{300}{4} = 75$) or when evicting an entry, the p_τ is therefore with higher r_c (i.e., $\frac{303}{3} = 101$) against the intended one (i.e., $\frac{200}{2} = 100$). Hence, the following theorem gives the necessary condition on $p(weight)$.

Theorem 4.2: $p(p_\tau, r_c)$ defines the entry lifetime (p_τ) to a specific PIT request (r_c). HLLR replaces the entry that has minimum r_c with highest p_τ value. Hence, the largest $p(weight)$ is calculated as follows:

$$p(weight)_{PIT} = \begin{cases} \mu_\tau & if\, r_c = 1 \\ max\left(\frac{PIT_{size} \cdot p_\tau}{r_c^{r_c+1} \cdot \sqrt[2]{\frac{Q_\tau}{\eta_R}}}\right)_p & otherwise \end{cases} \tag{4.27}$$

Proof: The above theorem shows a rule of choosing $p(weight)$ that has two options. Thus, by running the HLLR algorithm for managing PIT, when the PIT becomes overflowed and has a new Interest packet awaiting a schedule of insertion into the PIT, there will be a choice of selecting from the two available options mentioned above.

Using the proposed formal method concept of proximity for $p(weight)$, the $p(weight)$ can be more conveniently depicted with removal from PIT and decrease the lookup process time by denoting μ_τ to be a maximum PIT entry lifetime currently in the PIT for each PIT entry that has $r_c = 1$. Hence, the μ_τ is calculated by Equation (4.28):

$$\mu_\tau(PIT) = max(p_i(p_\tau))_{PIT} \tag{4.28}$$

where $p_i = \{p_1, p_2, p_3, ..., PIT_{size}\} \forall p_i \varepsilon PIT_{size}$. Hence:

$$p(weight)_{PIT} = \mu_\tau(PIT) \tag{4.29}$$

This means that the $p(weight)_{PIT}$ equals the maximum PIT entry lifetime for all entries in the PIT if the request of this entry equals one. By applying Equation (4.27) for the first option, it may enable the HLLR to search for the entry that has only one request and maximum lifetime to evict it from the PIT.

The second option is applied in order to find the accuracy entry based on the HLLR concept without any instability in the PIT when evicting an entry as earlier mentioned as a drawback of Equation (4.25). Therefore, the total number of requests (η_R) can be calculated for all entries inside the PIT that is expected as follows:

$$\eta_R(PIT) = \sum p_i(r_c) \tag{4.30}$$

where $p_i = \{p_1, p_2, p_3, ..., PIT_{size}\} \forall p_i \varepsilon PIT_{size}$, since η_R is a total number of requests sent by the subscribers and received from all faces. In addition, there is the need to calculate the total amount of time (Q_τ) which all entries are allowed to stay inside PIT, and this is to be acquired based on:

$$Q_\tau(PIT) = \sum p_i(p_\tau) \tag{4.31}$$

where $p_i = \{p_1, p_2, p_3, ..., PIT_{size}\} \forall p_i \varepsilon PIT_{size}$. Therefore, the probability of $p(weight)$ for one entry in PIT is given as:

$$p(weight) = \frac{PIT_{size} \cdot p_\tau}{r_C^{r_c+1} \cdot \sqrt[2]{\frac{Q_\tau}{\eta_R}}} \tag{4.32}$$

Hence, to look up and determine exactly the entry that should be evicted from the PIT, it is given by the following equation:

$$p(weight)_{PIT} = max(p_i(weight)) \tag{4.33}$$

where $p_i = \{p_1, p_2, p_3, ..., PIT_{size}\} \forall p_i \varepsilon PIT_{size}$.

In the first instance, HLLR is implemented and developed based on traditional NDN PIT. Thus, HLLR will be activated when the real PIT becomes overflowed. The proceeding algorithm of HLLR is presented in Algorithm 4.4:

Algorithm 4.4 Treatment of entry in HLLR when PIT overflow

Input:
p_τ: Entry lifetime
p: Incoming entry
PIT_{size}: Number of packets currently in the PIT
Main process:
Begin
if the size of PIT does overflow
  Check **if** there is any entry has one request **then**
    Find maximum PIT entry lifetime among them based on $\mu_\tau \leftarrow max(p_i(p_\tau))_{PIT}$
    Calculate the weight of entry for evicting it from PIT based on $p(weight)_{PIT} \leftarrow \mu_\tau$
  end_if
 Go to step #
    else // *PIT does no overflow*
      Counted the number of request for each entry existing in PIT
      Calculate the total number of request for all entries inside PIT based on $\eta_R(PIT) \leftarrow \sum p_i(r_c)$
      Calculate the total amount of time for all entries staying inside PIT based on $Q_\tau(PIT) \leftarrow \sum p_i(p_\tau)$
      Calculate the weight to each entry based on $p(weight) \leftarrow \frac{PIT_{size} \cdot p_\tau}{r_c^{rc+1} \cdot \sqrt[2]{\frac{Q_\tau}{\eta_R}}}$
      Calculate the price of entry for removing it from PIT based on $p(weight)_{PIT} \leftarrow max(p(weight)_{p_i|\forall p_i \varepsilon PIT_{size}}.)$
    end_else
#: Replacement $(p(weight)_{PIT}, p)$ in order to update PIT
end_if
End

4.4.3 HLLR Verification and Validation

The proposed policy (HLLR) was verified inside the ndnSIM simulator in order to ensure that the policy deploys those assumptions correctly. Verification was conducted following the techniques mentioned in Chapter Three. After the verification process, the following were confirmed:

- HLLR policy is programmed correctly, and

- HLLR policy implementation does not contain any errors or bugs.

To validate the accuracy of the proposed HLLR policy, two methods were employed, firstly by numerical analysis via executing one operation in order to ascertain how the HLLR policy works. The Interest packet lifetime denoted initial contention to be between 3000 ms to 4000 ms for instance. The NDN router was assumed to keep receiving Interest packets from each face and update PIT according to network behavior.

For example, the Table 4.4 shows 30 Interest packets that need to be cached in the PIT. Therefore, it was assumed that all Interest packets will be recorded or updated in the PIT, and none will be rejected. The maximum capacity of PIT was equal to 10 entries. Since the PIT is a data structure, each PIT entry contains the following information: *(i)* the name associated with the entry (E_name), *(ii)* a list of incoming faces (I_face), *(iii)* a list of outgoing faces (O_Face), *(iv)* the time when the entry should expire (LT), and *(v)* forwarding-strategy (FS).

Table 4.4
Interest Packet Data Set

Entity Name	Incoming Face	Lifetime	Enqueue PIT Time
www.google.com/	4	3196	1
www.onlinecorrection.com/	1	3131	2
www.ss.uni/	5	3269	3
www.powervoip.com/	3	3230	4
www.facebook.org/	1	3220	5
www.onlinecorrection.com/	2	3121	6
www.youtube.com	2	3182	7
www.tm.com.my/	1	3186	8
www.ss.uni/	4	3211	9
www.google.com/	3	3120	10
www.tm.com.my/	3	3161	11
www.lelong.com.my/	3	3239	12
www.tm.com.my/	4	3047	13
www.internetworks.my/	2	3321	14
www.onlinecorrection.com/	3	3094	15
www.uobabylon.edu/	1	3129	16
www.tm.com.my/	5	3131	17
www.internetworks.my/	1	3265	18
www.ss.uni/	1	3208	19
www.google.com/	5	3139	20
www.uobabylon.edu/	2	3072	21
www.youtube.com	3	3062	22
www.youtube.com	1	3055	23
www.tm.com.my/	2	3079	24
www.onlinecorrection.com/	4	3100	25
www.internetworks.my/	4	3262	26
www.facebook.org/	2	3281	27
www.youtube.com	1	3109	28
www.facebook.org/	4	3239	29
www.lelong.com.my/	1	3288	30

By executing one operation in order to ascertain how the current PIT replacement policies works, it is assumed that an NDN router keeps receiving the Interest packets from each face and updates PIT according to the network behavior. For example, a data set represents 30 Interest packet that needs to be serviced in the PIT. Also, it is assumed that all Interest packet will be recorded or updated in PIT and no one will be rejected, where the capacity of PIT is equal to 10 entries for instance. Since the PIT is a data structure, each PIT entry contains the information that is illustrated in Figure 2.10. In this regard, Figure 4.12 illustrates the PIT after the entries were recorded or updated based on these fields and assuming each entry lifetime is decreased by one when a new entry is recording or updating an existing one (e.g., the lifetime of "*www.google.com/*" entry is initially "3196 ms", and after 30 operations, the entry becomes "3166 ms"):

E_name	I_face	O_face	LT	FS
www.google.com/	4,3,5	1,2,6	3166	flooding
www.onlinecorrect.com/	1,2,3,4	1,5	3102	flooding
www.ss.uni/	5,4,1	2,3	3241	flooding
www.facebook.org/	1,2,4	3,5,6	3194	flooding
www.youtube.com	2,3,1,5	4,6	3158	flooding
www.tm.com.my/	1,3,4,5,2	6	3163	flooding
www.lelong.com.my/	3,1	5,2	3220	flooding
www.internetworks.my/	3,1,4	2,6,5	3304	flooding
www.powervoip.com/	3	1,5	3314	flooding
www.uobabylon.edu/	1,2	4	3114	flooding

Figure 4.12. PIT Entries

As described in Section 4.4.3, Equation (4.27) contains two options for updating PIT upon experiencing an overflow, which are shown in Figure 4.13 and Figure 4.14, respectively. By applying Equation (4.27) for the first option, it may enable the HLLR to search for the entry that has '1' frequency and maximum entry lifetime in order to remove from the PIT. In this case, the $p(weight)$ is 3314 ms because there is only one PIT entry has '1' frequency. Otherwise, when there are many entries have '1' frequency. The HLLR will be selected for the entry that has a maximum lifetime for replacing.

New entry record

www.slideshow.com/	1	2,5	3207	flooding

PIT is full - before treatment

E_name	I_face	O_face	LT	FS
www.google.com/	4,3,5	1,2,6	3166	flooding
www.onlinecorrect.com/	1,2,3,4	1,5	3102	flooding
www.ss.uni/	5,4,1	2,3	3241	flooding
www.facebook.org/	1,2,4	3,5,6	3194	flooding
www.youtube.com	2,3,1,5	4,6	3158	flooding
www.tm.com.my/	1,3,4,5,2	6	3163	flooding
www.lelong.com.my/	3,1	5,2	3220	flooding
www.internetworks.my/	3,1,4	2,6,5	3304	flooding
www.powervoip.com/	3	1,5	3314	flooding
www.uobabylon.edu/	1,2	4	3114	flooding

PIT - after treatment

E_name	I_face	O_face	LT	FS
www.google.com/	4,3,5	1,2,6	3166	flooding
www.onlinecorrect.com/	1,2,3,4	1,5	3102	flooding
www.ss.uni/	5,4,1	2,3	3241	flooding
www.facebook.org/	1,2,4	3,5,6	3194	flooding
www.youtube.com	2,3,1,5	4,6	3158	flooding
www.tm.com.my/	1,3,4,5,2	6	3163	flooding
www.lelong.com.my/	3,1	5,2	3220	flooding
www.internetworks.my/	3,1,4	2,6,5	3304	flooding
www.slideshow.com/	1	2,5	3207	flooding
www.uobabylon.edu/	1,2	4	3114	flooding

Figure 4.13. Entry Replacement based on HLLR, r_c=1

The second option for Equation (4.27) is applied in order to find the accurate entry based on the HLLR concept. The $p(weight)$ is calculated for each entry in PIT as represented in this set S={entry name, $p(weight)$}={(www.google.com/, 12.3648), (www.onlinecorrect.com/, 0.95830), (www.ss.uni/, 12.6577), (www.facebook.org/, 12.47420), (www.youtube.com, 0.97560), (www.tm.com.my/, 0.06403), (www.lelong.com.my/, 127.32943), (www.internetworks.my/, 12.90380), (www.powervoip.com/, 12.94286), (www.uobabylon.edu/, 123.13784))}. As we can see the entry (www.lelong.com.my/, 127.32943) that has maximum $p(weight)$, it should be evicted from PIT for the incoming Interest packet.

New entry record

www.slideshow.com/	1	2,5	3207	flooding

PIT is full - before treatment

E_name	I_face	O_face	LT	FS
www.google.com/	4,3,5	1,2,6	3166	flooding
www.onlinecorrect.com	1,2,3,4	1,5	3102	flooding
www.ss.uni/	5,4,1	2,3	3241	flooding
www.facebook.org/	1,2,4	3,5,6	3194	flooding
www.youtube.com	2,3,1,5	4,6	3158	flooding
www.tm.com.my/	1,3,4,5,2	6	3163	flooding
www.lelong.com.my/	3,1	5,2	3220	flooding
www.internetworks.my/	3,1,4	2,6,5	3304	flooding
www.powervoip.com/	3,4,2	1,5	3314	flooding
www.uobabylon.edu/	1,2	4	3114	flooding

PIT - after treatment

E_name	I_face	O_face	LT	FS
www.google.com/	4,3,5	1,2,6	3166	flooding
www.onlinecorrect.com	1,2,3,4	1,5	3102	flooding
www.slideshow.com/	3	2,5	3207	flooding
www.facebook.org/	1,2,4	3,5,6	3194	flooding
www.youtube.com	2,3,1,5	4,6	3158	flooding
www.tm.com.my/	1,3,4,5,2	6	3163	flooding
www.slideshow.com/	1	2,5	3207	flooding
www.internetworks.my/	3,1,4	2,6,5	3304	flooding
www.powervoip.com/	3	1,5	3314	flooding
www.uobabylon.edu/	1,2	4	3114	flooding

Figure 4.14. Entry Replacement based on HLLR, r_c>1

The second method to validate the accuracy of the proposed HLLR policy was by means of the simulation scenario. For this purpose, Sensitivity analysis technique is used by methodically changing the Interest packet rate and parameters of the HLLR across a range of interests and taking into consideration the effect on the behavior of the model. The flooding route was used as the routing protocol in this scenario. Additionally other system parameters include: Interest packet lifetime denoted initial contention of 4000 ms, Interest packet rate ranged from 100 to 1000 packets/second, PIT size was set to 100 entry packets, and CS size accommodated 100 entries. Thus, these scenario runs may show which policy provides a better performance. For each simulation scenario, the Interest packet retransmission rate values for HLLR policy obtained from the simulation experiments were compared with the Interest packet re-transmission rates for Persistent policy obtained from another simulation.

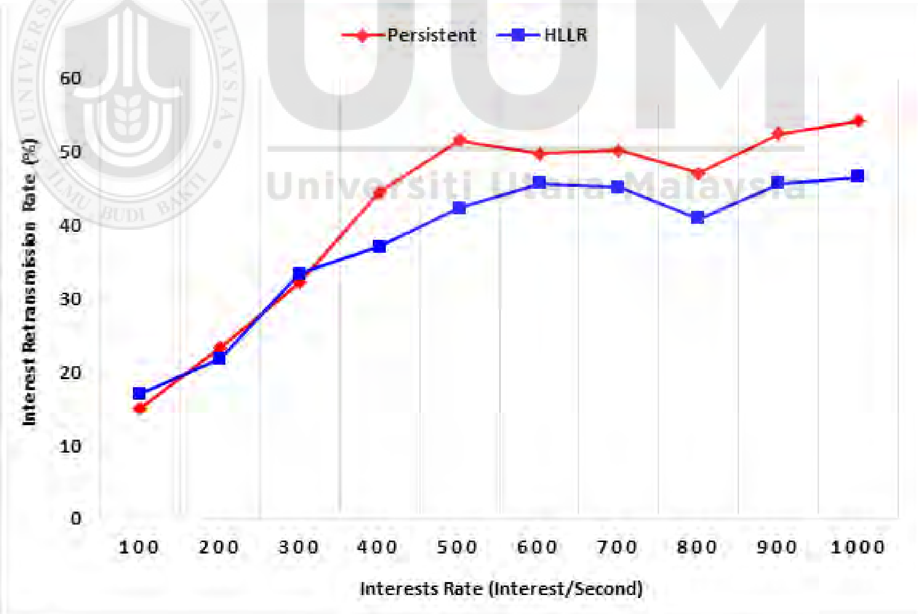


Figure 4.15. IRR vs Interests Rate for Persistent and HLLR

Figure 4.15 presents the Interest retransmission rate for Persistent and HLLR policies under varying Interest packet rates. The horizontal an x-axis plots the ten different Interest packet rates per second, while the vertical an y-axis plots the Interest retransmission rate. According to the results obtained, all policies share the same trend of increasing Interests packet retransmission in line with Interest packet rate increase. The graph surmises that when HLLR is applied, the drop packets are minimal and required retransmissions are at a minimum, thus saving the Interest packet retransmission cost. When the Persistent policy was applied, it gave a worse result since this policy does not take into account the PIT condition. This method drops the incoming Interest packet directly when the PIT is full.

4.4.4 HLLR Evaluation

In order to guarantee the consistency of the presented results, the scenarios were tested using ndnSIM simulator using the Abilene topology as shown in Figure 2.16. Least Recently Used, Random, and Persistent policies were applied as a replacement policy in order to evaluate HLLR policy performance. The Interest packet rate ranged from 250 to 2000 Interest/second, and for the PIT, the size was set to 100 entries, whereas the CS size was set to 100 entries. Flooding strategy was set as a forwarding strategy in this experimental demonstration. For each simulation scenario, the experiment was repeated 10 times as well as the simulation time set at 300 seconds. In particular, a comprehensive simulation was performed by measuring the Interest drop rate and Interest satisfaction rate usage via different number of scenarios.

Interest drop rate can usually be noticed due to network congestion, such as PIT overflow, queue overflow, and poor channel quality. The aim of this was to analyze the impact of Interest packet rate on the performance of network regarding the Interest drop performance metrics. Interest drop refers to the number of Interest dropped in the PIT because of the Interest packet replacement or Interest packet lifetime having expired. According to the results shown in Figure 4.16, it depicts that the disparity in Interest drop is very large between the LRU and Random, and Persistent and HLLR. Each policy condition differs in the form of dealing with the PIT when it becomes overflowed, particularly when the Interest packet rate is increased and there is no available space for entries to wait in the PIT memory to be satisfied. Therefore, it can be concluded that the Interest drop rate of HLLR is better as compared to the LRU, Random, and Persistent replacement policies. This is due to the fact that HLLR can provide better priority to entries with consideration of the lowest request with highest lifetime that needs to be removed to allow a new entry to be inserted.

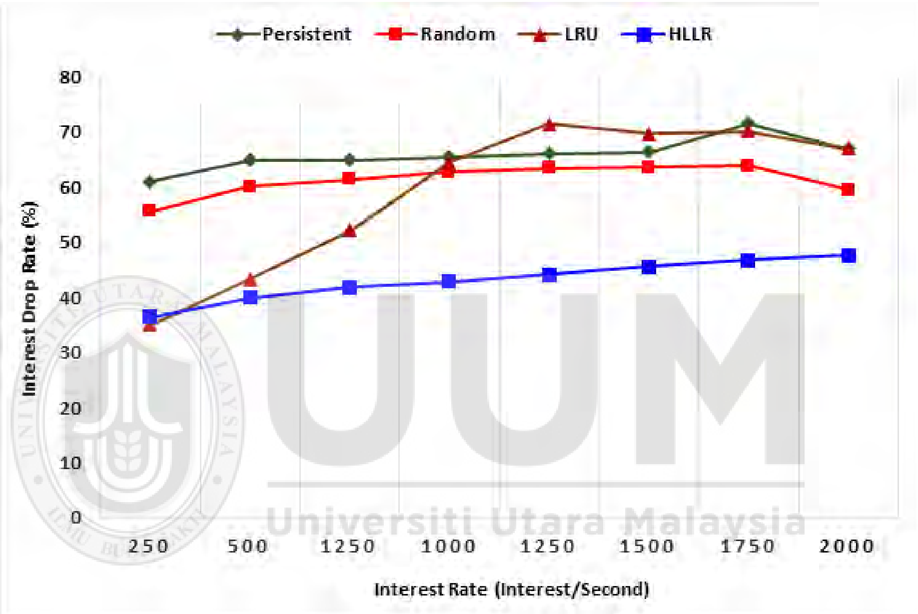


Figure 4.16. IDR vs PIT Size for Persistent, Random, LRU and HLLR

Figure 4.17 shows Interest satisfaction rate that refers to the percentage of satisfied Interest packets per NDN router in the network. The horizontal x-axis represents the eight different Interest packet rates per second, while the vertical y-axis represents the Interest packet satisfaction rate. In the scenarios where the Interest packet rate was increased, the Interest satisfaction rate returned almost the same behaviors by decreasing the Interest satisfaction rate, which implies that HLLR policy may increase the total Interest satisfaction rate. Hence, this guarantees that with a low and a high Interest packet rate, the Interest satisfaction rate is dramatically maximized as compared to

Persistent, Random, and LRU policies.

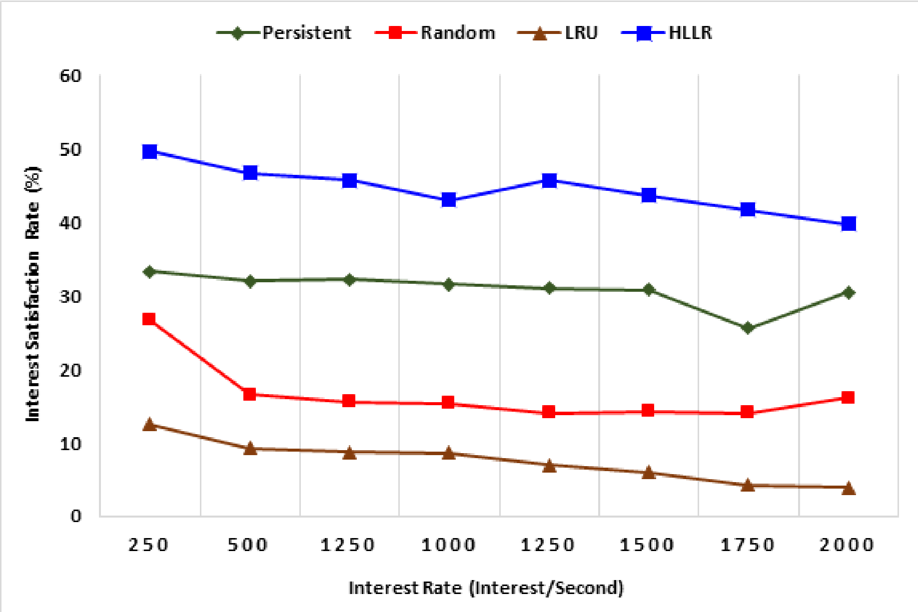


Figure 4.17. ISR vs Interest Rate for Persistent, Random, LRU and HLLR

As a conclusion, HLLR, Persistent, Random, and LRU were compared by using the Abilene topology and the results are shown in Figures 4.16 and 4.17. The analysis revealed that HLLR is capable of significantly reducing the Interest drop rate when compared with Persistent, Random, and LRU policies. In addition, HLLR achieves a higher Interest packet satisfaction at routers when compared with Persistent, Random, and LRU. Hence, the results supported that the HLLR policy can approximate and even outperform Persistent, Random, and LRU policies.

4.5 Summary

This chapter introduced the AVPIT model, the STIL model, and the HLLR model for PIT over a content-centric environment. The proposed AVPIT mechanism as a first contribution, was designed for indicating the impending overflow by using VPIT indicators. The second contribution is the STIL mechanism designed to adjust the Interest packet lifetime in order to avoid overstaying of PIT entries during heavy network load.

The HLLR policy as the third contribution, was designed to determine which PIT entry is to be replaced for efficient PIT utilization. In addition, these models were injected into traditional NDN via simulation in ndnSIM. The implications of these simulations were presented in 4.2, 4.3 and 4.4 sections of this chapter.

CHAPTER FIVE
SIMULATION ANALYSIS AND EVALUATION

The validation of Pending Interest Table Control Management (PITCM) approach, as presented in Chapter Four, has yielded positive results. In this chapter, the performance evaluation was presented to investigate the benefits of the features introduced by the PITCM approach through extensive simulations obtained by means of varying parameters and different topologies. For a better understanding and confirmation of the PITCM, the PITCM's performance was compared with that of traditional NDN PIT. Hence, this chapter starts with an overview and implementation of the structure of PITCM in Section 5.1. Results of the performance evaluations of PITCM over the traditional NDN are discussed in Section 5.2. The discussion of the performance evaluation is elaborated more in Section 5.3. Finally, a summary of the chapter is presented in Section 5.4.

5.1 Pending Interest Table Control Management

The ultimate aim of this thesis was to develop and implement an PITCM approach for NDN. The PITCM monitors and controls an incoming Interest packet as well as manages the PIT entries that deals with PIT overflow, which leads to increase the PIT utilization. In addition, the PITCM adopted an Negative ACKnowledgment (NACK) feedback operation at the downstream node when the PIT state overflows in the NDN router. Also, Additive Increase Multiplicative Decrease (AIMD) scheme was adopted to increase and decrease the Interest packet sending rate at the subscribers' nodes. PITCM incorporates three proposed components, namely AVPIT, STIL and HLLR, as presented in Chapter Four, to enhance the management operations of PIT in NDN router. Moreover, the PITCM makes several changes and extensions at the NDN's PIT to cope with the new behaviors of the heavy load of Interest packet that arrives to

NDN router. The research problem, goals specification, and research focus contributed highly in designing the conceptual framework of PITCM.

AVPIT, STIL and HLLR were compiled at PITCM as a one approach and integrated into NDN router code, which was implemented in the ndnSIM environment. Incorporating PITCM components aim to provide a fundamental direction of new PIT management in NDN router, which leads to enhance the PIT performance significantly as well as enhance the entire network performance. Each step in the PITCM pseudo-code (see Algorithm 5.1) addresses a particular issue in the PIT, which is described as follows. For each incoming Interest packet, when the Interest packet arrival epoch to PIT do, VPIT situation is checked whether the VPIT is overflowed or not. A positive result (VPIT does not overflow), in this case, the size of PIT and the size of VPIT are updated by increasing them. In addition, the NDN router grants larger lifetime values for incoming Interest packets (calling the STIL mechanism). Continuously with normal NDN processing by adding the incoming entry into PIT.

Otherwise, a negative result (VPIT does Overflow), NACK packet will be sent to the subscriber nodes to inform them about the PIT status in order to decrease the Interest packet sent rate. In the same time, the NDN router was updated the lifetime value for incoming Interest packet by grants shorter lifetime value to it (calling the STIL mechanism). In addition, PITCM will be activated the feedback method in order to send a NACK packet to the subscribers to inform them about the PIT status to be full, and thus it needs to decrease the Interest rate. Based on that, subscribers will decreased the Interest packets rate via using AIMD scheme. Furthermore, the PITCM will be also activated HLLR replacement policy in order to determine PIT entry to be changed with the new incoming Interest packet for efficient PIT utilization. With this approach, we avoid any additional traffic into the network and network resource.

Algorithm 5.1 PITCM Processing

Input:
p : Interest packet
PIT_{size} : Number of packets currently in the PIT, initially equal 0
$VPIT_{size}$: Number of packets currently in the VPIT, initially equal 0
η_{PIT} : Limit size of PIT
T_{th} : Threshold lifetime of PIT, initially equal 4000 Or based on subscribers setting
p_{τ}: Interest lifetime, initially by subscribers
VPIT_condition : False

Main process:
Begin
////////////////// At each Interest packet arrival epoch to PIT do
Update the size of VPIT, calculating based on $VPIT_{size} \leftarrow max(VPIT_{size} - \check{C}.t)$
$VPIT_{size}$ is checked with incoming Interest packet size **if** it is less than PIT limit size **then** // *PIT does not overflow*
PIT_{size} is updated based on $PIT_{size} \leftarrow PIT_{size} + p$
$VPIT_{size}$ is updated based on $VPIT_{size} \leftarrow VPIT_{size} + p$
VPIT_condition ← false
Call STIL(*VPIT_condition*, p_{τ}, p, T_{th})
A new Interest will be added to the PIT // *normal NDN processing*
end_if
else // *PIT does overflow*
Return Interest NACK (PIT_Overflow) to subscriber nodes
VPIT_condition ← true
Call STIL(*VPIT_condition*, p_{τ}, p, T_{th})
Call HLLR (*VPIT_condition*, p_{τ}, p, PIT_{size})
end_else
End

To carefully evaluate the PITCM approach, the proposed solutions have been implemented in the official NDN simulator (i.e., ndnSIM). The simulations were carefully set to match the specification of this study. Therefore, the evaluation has been conducted to study the impact of the PITCM approach on the performance of PIT based on AVPIT, STIL and HLLR as well as to compare the performance of the proposed approach with the traditional NDN PIT.

PITCM-test scenario, output results snapshot are illustrated in Figures 5.1 and Figure 5.2. In addition, the researcher will offer assistance and advice on the implementation issues of PITCM in any environment besides the ndnSIM simulator.

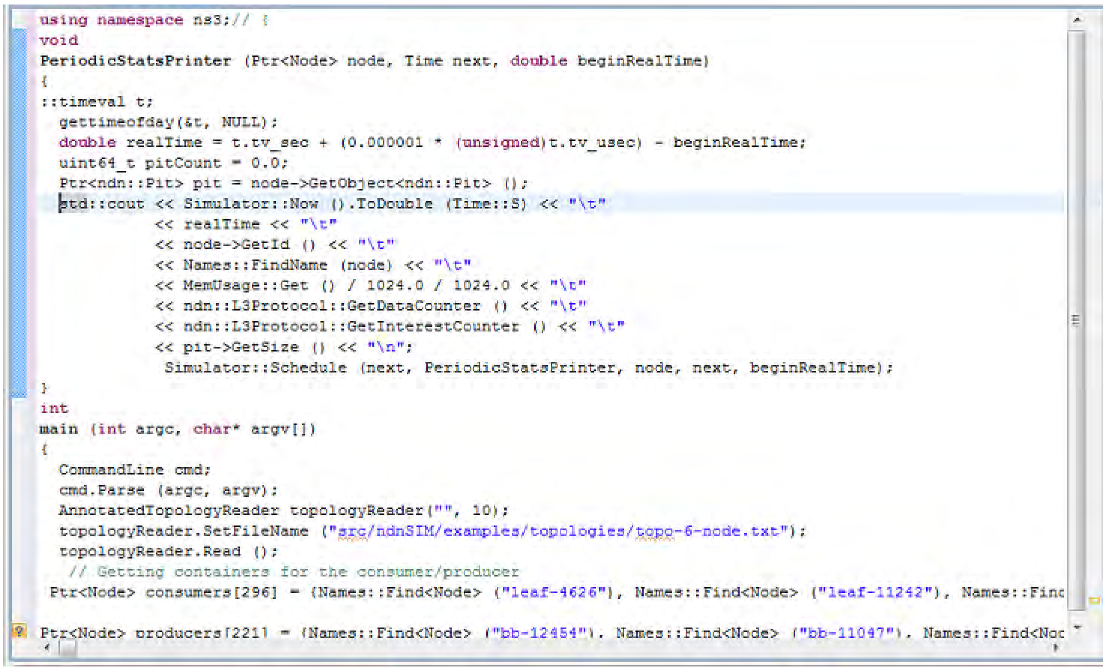


Figure 5.1. PITCM: Test Scenario

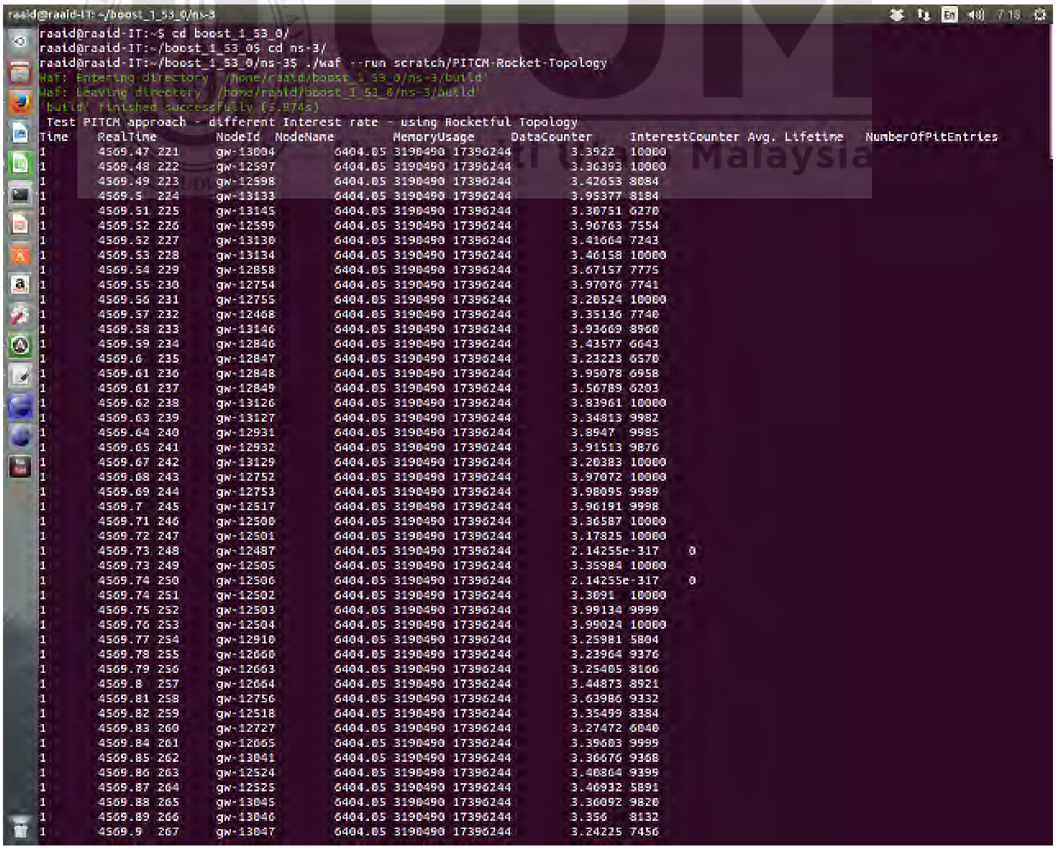


Figure 5.2. PITCM: Output Results Snapshot

5.2 Performance Evaluation of PITCM

The evaluation is the main part of our methodology and choosing the evaluation method is a very significant step in this stage for any research or project [181], particularly when the comparative analysis is needed among several designs to present which design outperforms the others [308]. In this study, simulation method will be used due it is particularly used for evaluation of many reviewed techniques. Choosing the realistic simulation tools is very essential to develop meaningful experiments and carry out the real behaviors for the proposed protocol (approach) in different phases [211].

The main goal of these evaluations is to test the ability of the PITCM to manage the PIT entries in NDN as compared to the traditional NDN PIT. The attempt was not to measure the PITCM's performance on a particular workload captured from a real network, rather, it was to measure its performance under a range of network conditions and scenarios. For the fair evaluation of PIT issue, there is no general agreement on the particular topology selection [214]. Therefore, the focus was set on some network topologies, i.e., Abilene (small topology) and Rocketfuel-mapped AT&T (large scale topology) with a variety of Iterest packet rate and PIT size.

The key step in all performance evaluation parts are the performance metrics selection [318]. However, different researchers consider the performance metrics can mean different things depending on the situation in which they are used [306]. Thus, selecting the proper performance metrics is very important to investigate the behaviors of the proposed approach from a different perspective. In addition, different metrics used in different scenarios with varying nodes density give us the complete view on the performance of the proposed mechanism. In particular, this study focuses on the PIT length, Interest retransmission rate, Interest drop rate, Interest satisfaction rate and

Interest satisfaction delay metrics which could be used to measure the performance of the proposed objectives of this research.

5.2.1 Abilene and Rocketfuel Topologies with Various Interest Rate

PITCM approach and conventional NDN PIT were tested under different Interest packet rate: 1000 to 10000 Interest/second; Moreover, the PIT size which specifies the available space in every node for temporarily storing entries, the setting is 1000 entries, and for CS size, the setting is 10000 entries. PITCM and conventional NDN PIT were tested all the scenarios on Abilene topology and Rocketfuel-mapped topology. The detailed information of the simulation configuration and parameter values was provided in Table 3.2. For each simulation scenario, the experiment is repeated several times and the average value is obtained. In addition, for evaluating the behavior of PITCM, five metrics were selected which are PIT length, Interest retransmission rate, Interest drop rate, Interest satisfaction rate and Interest satisfaction delay.

5.2.1.1 Abilene Scenarios

This work intends to analyze the performance impact of the traffic load on the PIT occupancy. Thus, to guarantee the consistency of the presented results, the PITCM approach was simulated on the Abilene scenario (see Figure 2.16). Abilene consists of 11 NDN routers, 25 subscriber nodes and 10 publisher nodes. This study intends to analyze the performance impact of Interest packet rate on the occupancy of PIT. Therefore, the link capacities and delays were set as follows: the capacities and delays of subscriber to NDN router links are set as 0.01 Gbps and 10 ms, respectively; NDN router to NDN router links are set as 0.005 Gbps and 10 ms, respectively and NDN router to publisher links are set as 0.01 Gbps and 10 ms, respectively. Each subscriber/publisher is connected to one NDN router uniformly selected between those available in the network.

Figure 5.3 shown the PIT length of traditional NDN PIT and PITCM. In the given figure, an x-axis shows the Interest packet rate per second, whereas an y-axis represents the PIT length. The PIT length refers to the amount of non-expired pending Interests existing on each NDN router on the topology. Both NDN PIT and PITCM experienced a considerable PIT length increase as the Interest packet rate increases. When the Interest packet rate is equal to 1000 Interest/second, the PITCM PIT length is equal to (623 entries). This is followed by a dramatic increase as the Interest packet rate increase. Then, PIT length stabilized at about (760-732 entries) with the Interest packet rate being more than 5000 Interest/second. Over the same Interest packet rate, the PIT length of the NDN PIT started at (806 entries) with 1000 Interest packet rate per second and a considerable increase dramatically to (986 entries) at 10000 Interest packet rate per second. In brief, PITCM achieves PIT length lower than NDN PIT over all presented Interest packet rate. Remarkably, the performance gap between the PITCM and the NDN PIT increases with the Interest packet rate to reach almost (34.96% entries) when the Interest packet rate is 10000 Interest/second.

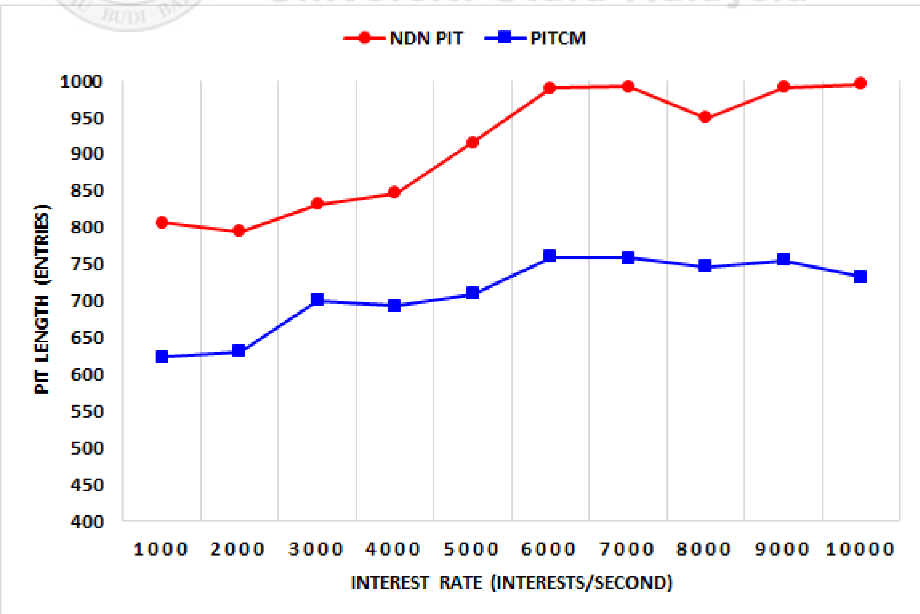


Figure 5.3. Abilene Topology: PIT Length vs Interest Rate for NDN PIT and PITCM

The Interest retransmission rate refers to the amount of Interest packet retransmissions when it is reissued because of timeouts or drops. Figure 5.4 summarizes the Interest retransmission rate of PITCM and NDN PIT. In the given figure, an x-axis presents the Interest packet rate per second, whereas an y-axis represents the Interest retransmission rate. According to the results obtained, it can be seen for the PITCM, the Interest retransmissions is increased as Interest packet rate increases, which are achieved between (9.12%) to (17.53%) with Interest packet rate from 1000 to 10000 Interest/second. On the other hand, NDN PIT obtained same behavior as a results since the Interest packet rate increases. The Interest retransmissions also increase incremental, which is achieved between (10.11%) to (26.59%) with an Interest packet rate from 1000 to 10000 Interest/second. Over all, the PITCM performed better than NDN PIT, and it also had a value competitively lower than NDN PIT with a heavy workload. The result therefore showed that the PITCM reduced the Interest retransmission rate from (10.82% to 51.64%) at Interest packet rate 1000 to 10000 Interest/second, respectively.

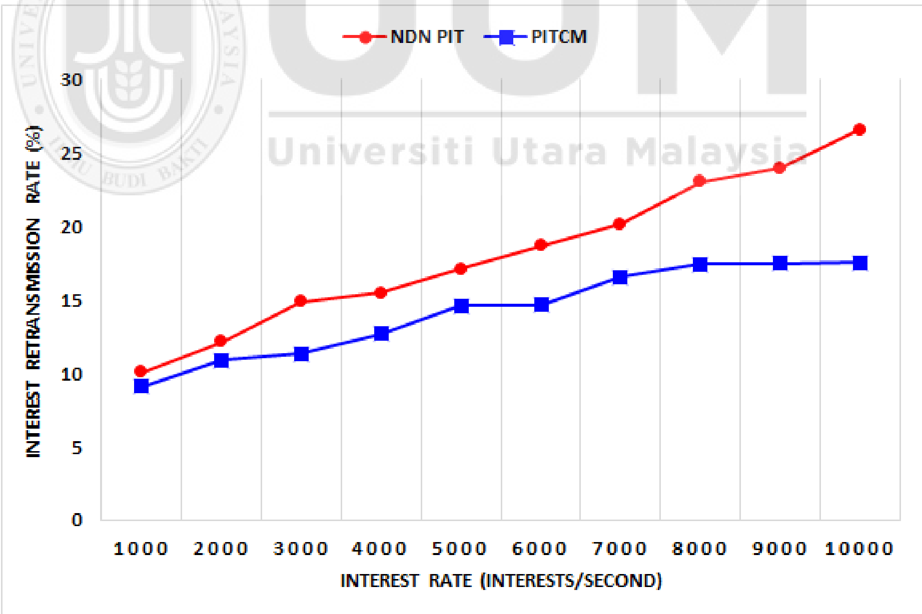


Figure 5.4. Abilene Topology: IRR vs Interest Rate for NDN PIT and PITCM

Figure 5.5 shows the Interest drop rate of PITCM and NDN PIT, which means the amount of Interest drops because the PIT impending overflow. In the given figure, an x-axis presents Interest packet rate per second, whereas an y-axis represents the

Interest drop rate. The result plots the relationship between the tested PITCM and NDN PIT in terms of Interest drop rate and how it can impact the behavior of PIT performance as well as the entire networking. PITCM is observed to clear the high traffic experienced by NDN PIT. Hence, by increasing the Interest packet rate 4000 Interest/second, a high impact is displayed with a high-Interest drop rate, which is somehow (15.11%). In NDN PIT, this percentage is increased to (22.65%) since the Interest packet rate is 3000 Interests/second, and continues to increase. However, the PITCM is recorded a better/decrees Interest drop rate score compared with the NDN PIT that it is reached around (41.92%) when the Interest packet rate is 10000 Interest/second.

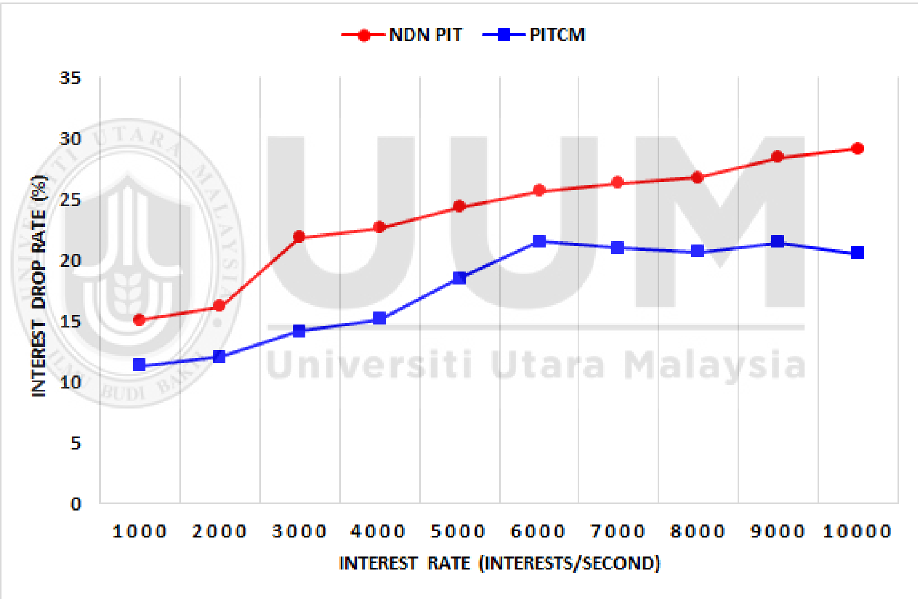


Figure 5.5. Abilene Topology: IDR vs Interest Rate for NDN PIT and PITCM

One of the main concepts of NDN is that the packets must be forwarded with minimum delay by making the NDN routers have the ability to store the copy from every Data packet passing over these nodes. However, PIT overflow is in sometime another reason to introduce delay when Data packet return and did not find any matching with the PIT entries due the entry is dropped or replaced. Therefore, it is important to measure the delay when designing a new approach. Figure 5.6 summarizes the Interest satisfaction delay of PITCM and NDN PIT over ten different Interest packet rates. The Interest satisfaction delay refers to the time interval between the first Interest generated by the subscriber for the specified content and the Data packet received in satisfying. Although NDN PIT and PITCM showed a gradual rise in Interest satisfaction delay as the Interest packet rate rises, NDN PIT seemed to have Interest satisfaction delay higher than PITCM. In brief, PITCM achieves between (23.27%) and (51.45%) lower than NDN PIT Interest satisfaction delay as the Interest packet rate is 1000 to 10000 Interest/second, respectively.

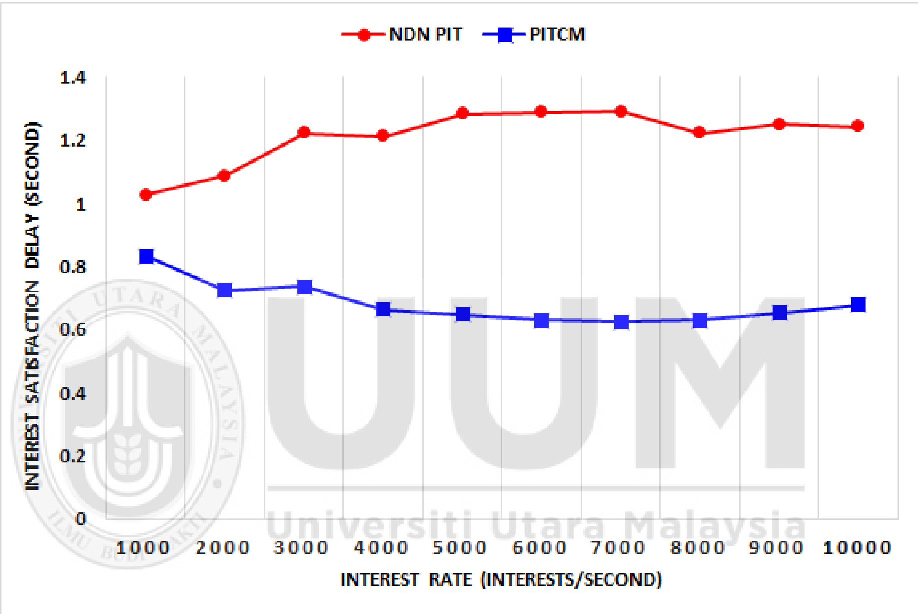


Figure 5.6. Abilene Topology: ISD vs Interest Rate for NDN PIT and PITCM

Figure 5.7 depicts the performance comparison of PITCM and NDN PIT in terms of Interest satisfaction rate, which is the number of Interest packet that satisfied per NDN router in the network. In the given figure, an x-axis presents Interest packet rate per second, whereas an y-axis represents the Interest satisfaction rate. According to the results obtained, PITCM and NDN PIT may share the same line since they decrease the Interest satisfaction as the Interest packet rate increases. However, as seen in the given figure, with the increase of the Interest packet rate from 1000 to 10000, the Interest satisfaction rate is recorded a decrease from (83.08%) to (55.37%) in NDN PIT. In PITCM, this reduction is from (82.89%) to (71.93%). Hence, PITCM guarantees that with high Interest packet rate scenarios (10000 Interest packet rate per second), the Interest satisfaction rate can be maximized, which is (29.89%), as compared to NDN PIT.

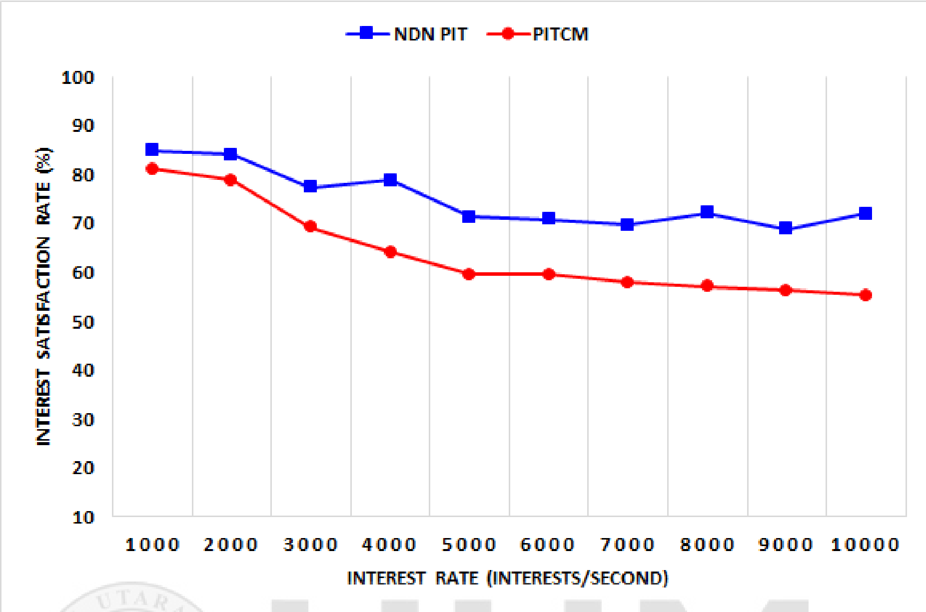


Figure 5.7. Abilene Topology: ISR vs Interest Rate for NDN PIT and PITCM

In conclusion, NDN PIT and PITCM were compared in the Abilene topology, and the results are presented in Figures 5.3 to 5.7. The analysis shows that both NDN PIT and PITCM operate similarly in dealing with the various Interest packet rate and PIT conditions. PITCM achieves a higher Interest satisfaction rate when compared with NDN PIT. In addition, PITCM reduces significantly the Interest retransmission rate, the Interest drop rate and the Interest satisfaction delay experienced by the Interest packet rate. Moreover, PITCM is able to maintain an PIT length at a high Interest packet rate.

5.2.1.2 Rocketfuel Scenario

In order to model a more realistic network environment, PITCM evaluated over a modified version of a Rocketfuel-mapped AT&T topology (i.e., large scale simulations). Rocketfuel-mapped topology nodes is separated into three types: subscribers (296), NDN routers (108), and publishers (221) nodes. In order to analyze the performance impact of the traffic load on the PIT occupancy. Therefore, the link capacities and delays were setting as follows: the capacities and delays of subscriber to NDN router links are set as 0.1 Gbps and 10 ms, respectively; NDN router to NDN router links are set as 0.05 Gbps and10 ms, respectively and NDN router to publisher links are set as 0.1 Gbps and 10 ms, respectively. In addition, each publisher or subscriber is connected to one NDN router uniformly selected between those available in the network. Hence, this section aims to evaluate the performance of PITCM relative to current NDN PIT under various Interest packet rate.

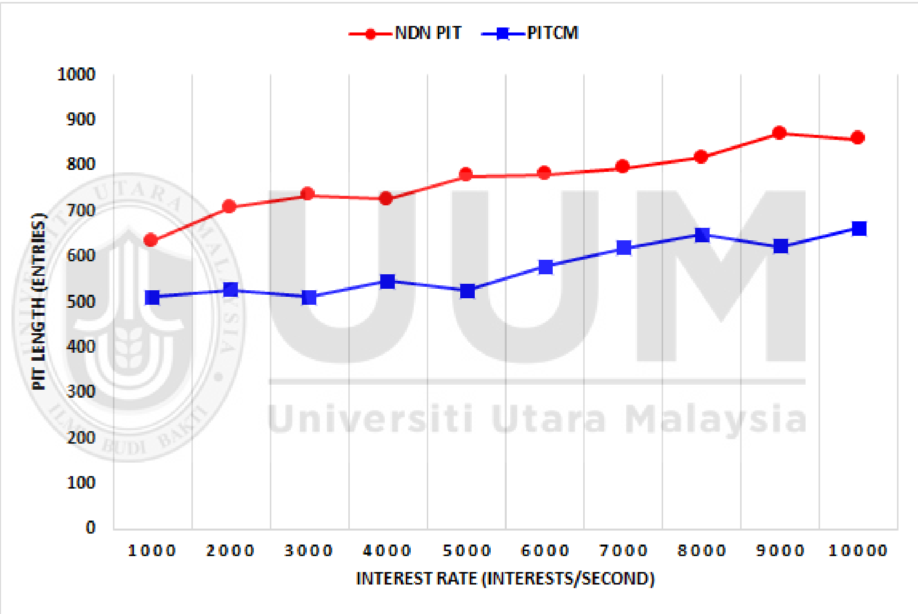


Figure 5.8. Rocketfuel Topology: PIT length vs Interest Rate for NDN PIT and PITCM

Figure 5.8 compares the PIT length of PITCM and NDN PIT over ten different Interest rates. As far as PITCM is concerned, the PIT length increased significantly as the Interest packet rate increases. PIT length is (512 entries) at 1000 Interest packet rate per second, and increases to (663 entries) at 10000 Interest packet rate per second. By contrast, NDN PIT length suffers a significant overflow at the same Interest packet rates from approximately (634 entries) to (858 entries) at 1000 and 10000 Interest packet rate per second, respectively. Anyhow, the graph shows that PITCM achieves a considerable increase over NDN PIT from (23.8% entries) at 1000 Interest packet rate per second to (29.32 entries) at 10000 Interest packet rate per second.

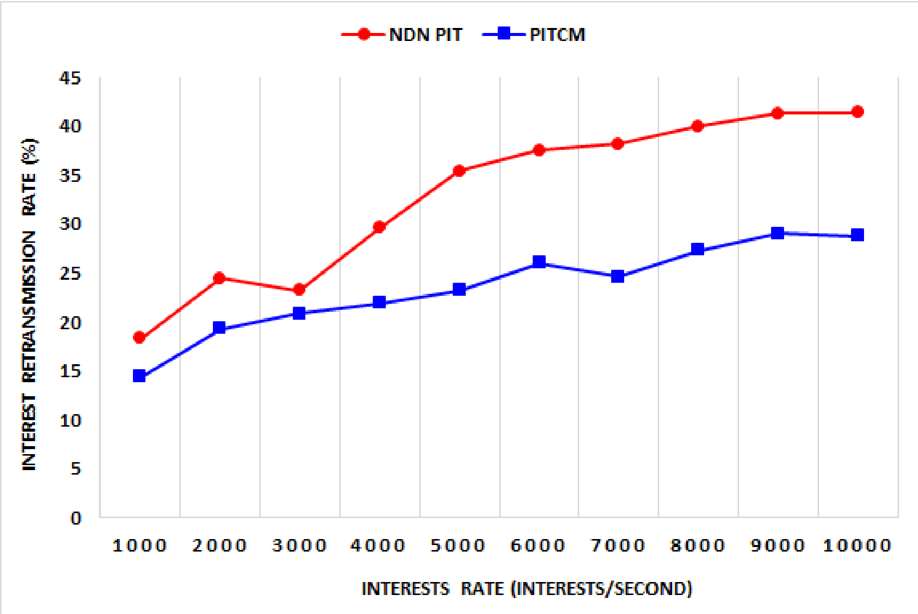


Figure 5.9. Rocketfuel Topology: IRR vs Interest Rate for NDN PIT and PITCM

Figure 5.9 depicts the Interest retransmission rate of PITCM and NDN PIT over ten different Interest packet rates. The Interest retransmission of both PITCM and NDN PIT have been increased linearly as the Interest packet rate increases. However, the Interest retransmission of NDN PIT has a severe altitude, particularly in 1000 and 10000 Interest packet rate per second. Thus, the Interest retransmission fell dramatically to (18.27%) and (41%) as compared to PITCM (14.39%) and (28.76%). Remarkably, PITCM achieves (43.82%) better/lower Interest retransmission than NDN PIT at 10000 Interest packet rate per second.

Figure 5.10 shows the Interest drop rate of PITCM and NDN PIT over ten different Interest packet rates. Overall, NDN PIT Interest drop is much greater than PITCM in all scenarios. The gap is great in the high-Interest packet rate, and it reduces in a low Interest packet rate. NDN PIT Interest drop fluctuates between 5000 Interest packet rates per second as (55.36%) and 10000 Interest packet rate per second as (71.37%),

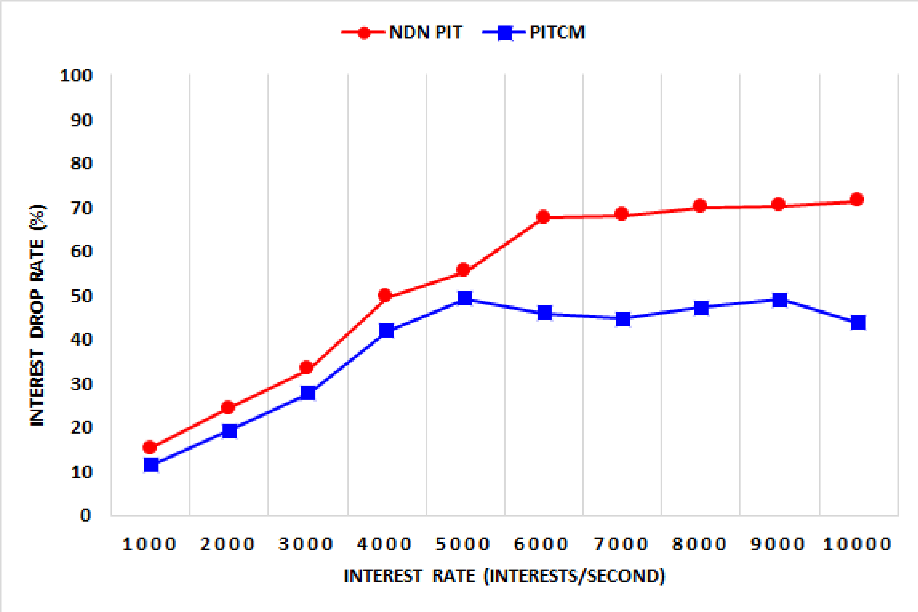


Figure 5.10. Rocketfuel Topology: IDR vs Interest Rate for NDN PIT and PITCM

whereas PITCM Interest packet rate is 5000 Interest/second, the Interest drop increases gradually to become (49.21%) and start to decreases to become (43.76%) when the Interest packet rate is 10000. To conclude, PITCM moderately adapts the growth of the Interest packet rate as a result by using the proposed AVPIT mechanism and STIL mechanism. In contrast, NDN PIT controller could not control the growth of Interest packet that arrives at the PIT.

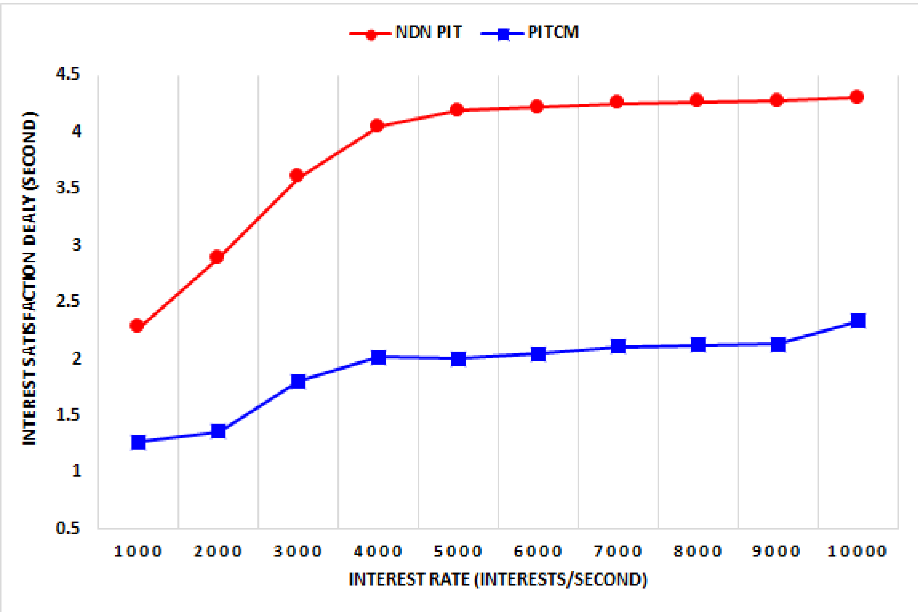


Figure 5.11. Rocketfuel Topology: ISD vs Interest Rate for NDN PIT and PITCM

Figure 5.11 illustrates the Interest satisfaction delay of PITCM and NDN PIT over ten different Interest packet rates. For evaluating the delay faced by Interests packet that generated during the simulations, it is observed that the Interest satisfaction delay linearly increases with the increase in the number of the Interest packet rate provided by the subscriber. Thus, PITCM achieves between (78.95%) and (84.14%) lower than NDN PIT Interest satisfaction delay at 1000 and 10000 Interest packet rate per second, respectively. As a result, PITCM faces less delay during the content retrieval process as compared to the NDN PIT.

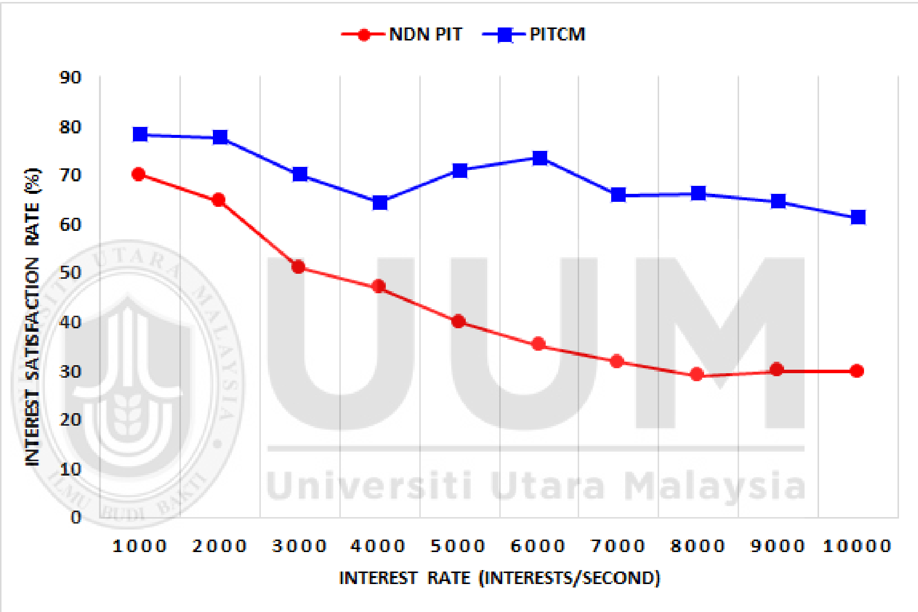


Figure 5.12. Rocketfuel Topology: ISR vs Interest Rate for NDN PIT and PITCM

To obtain a deeper understanding of the PITCM performance, Figure 5.12 illustrates the Interest satisfaction rate of PITCM and NDN PIT over ten different Interest packet rates. As a result, it can be seen that the PITCM Interest satisfaction has increase compared to the traditional NDN PIT. Hence, when the Interest packet rate is equal to 1000 Interest/second, the Interest satisfaction of PITCM is (78.24%) compared to (70.01%) in the case of NDN PIT. The difference increases as the Interest packet rate increases continuously. However, the difference between PITCM and NDN PIT becomes stabilized and reaches the value (105%) in 10000 Interest packet rates per second.

PITCM and NDN PIT were compared by varying the Interest packet rate, and the metrics results were measured including the PIT length, Interest retransmission rate, Interest drop rate, Interest satisfaction delay and Interest satisfaction rate presented in Figures 5.8 to 5.12. Thus, PITCM performs better than NDN PIT with low and high Interest packet rate. Simulation results show the effectiveness of this approach in both reducing PIT length, Interest satisfaction delay, Interest retransmission rate, Interest drop rate, and increasing interest satisfaction rate for legitimate subscribers. Thus, PITCM approach may mitigate the collapse that can occur in today's Internet when an Interest packet is dropped because of the PIT overflow and the repeated retransmissions from the subscriber node(s) that consume most of the bandwidth. Hence, for a heavy load Interest rate, PITCM is preferred more than NDN PIT.

5.2.2 Abilene and Rocketfuel Topologies with Various PIT Size

Although the theoretic analysis gives us a good understanding of how PIT should be managed to achieve better performance. However, In order to have reliable results, it important to design a wide range of possible scenarios. In addition, to fully understand the PIT sizing effect, we varied PIT size (capacity) from small (1000 entries) to large (10000 entries), which allows us to see the impact (if any) of even large PIT size in the PITCM approach. For that, ndnSIM simulator has chosen as a performance evaluation technique for this research, as mentioned before. It had been used successfully for implementing PITCM approach and tested it. PITCM is tested all the scenarios on Abilene and Rocketfuel topologies.

To analyze the size of PIT effect, the maximum number of entries can be stored were varied as 1000, 2000, 4000, 8000, and 10000 entries and for CS size is by setting

10000 entries. Moreover, the subscriber Interest rate is 10000 per second. The detailed information of the simulation configuration and parameter values is provided in Table 3.2. For each simulation scenario, the experiment is repeated several times and the average value is obtained. In order to evaluate the behavior of PITCM, five metrics are selected, which are PIT length, Interest retransmission rate, Interest drop rate, Interest satisfaction delay and Interest satisfaction rate.

5.2.2.1 Abilene Scenario

In Abilene topology where an NDN node represents a source and an associated cache. Sources may route requests to neighboring caches. Therefore, this section introduces the next level of performance PITCM approach, i.e., performance evaluation with simulation under varying size of PIT.

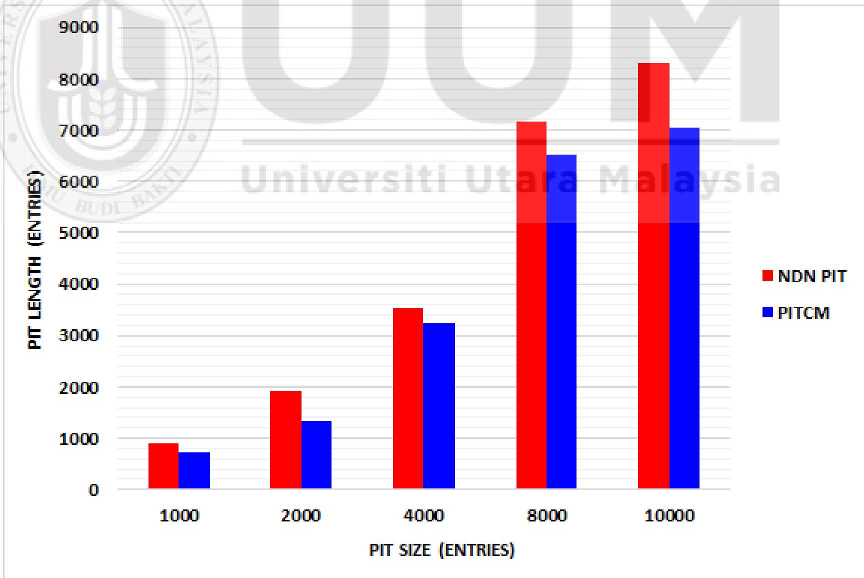


Figure 5.13. Abilene Topology: PIT Length vs PIT Size for NDN PIT and PITCM

Figure 5.13 plots the PIT length and the different PIT size. It is noted here that both NDN PIT and PITCM experienced a considerable PIT length increase as the PIT size increases. Over all the results, PITCM achieves lower PIT length than NDN PIT over all presented PIT size, which is because the PITCM holding optimal managing of all incoming Interest in their PIT. When the PIT size is setting to 2000 entries, the PIT length of PITCM is equal to (1328 entries). After that, there is a dramatic increase as the PIT size setting increase to (7049 entries) with PIT size is setting as 10000 entries. On the other hand, the PIT length of NDN PIT started at (1926 entries) with 2000 PIT size and considerable increase dramatically to (8313 entries) at 10000 PIT size. Noteworthy, the performance gap between PITCH and NDN PIT increased with the PIT size decreased to reach almost (45.04%) at 2000 entries of PIT size.

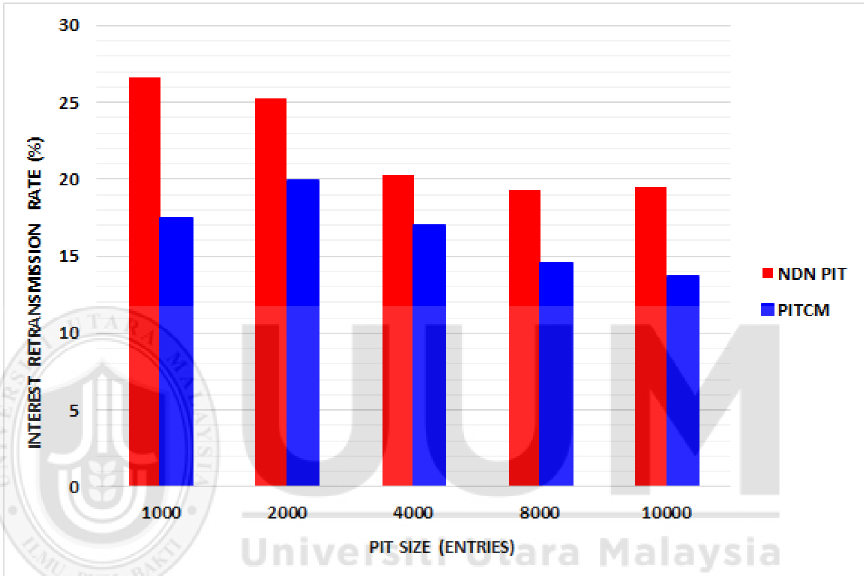


Figure 5.14. Abilene Topology: IRR vs PIT Size for NDN PIT and PITCM

As showed in Figure 5.14, the Interest retransmission rate over the five PIT sizes is presented for the PITCM and NDN PIT. As seen in the given figure, when PIT size is setting 2000 entries, the gap between PITCM and NDN PIT is increased as (26.47%), because of the ability of PITCM to reduce the Interest rate from a downstream node when the impending PIT overflow. Similarly, when the PIT size setting is 10000 entries, the Interest retransmission for PITCM records is decreased as (13.73%). In NDN PIT, this decrease is observed as (19.47%). Over all, the PITCM performed better than NDN PIT, and it also had a value competitively lower than NDN PIT with a small PIT size.

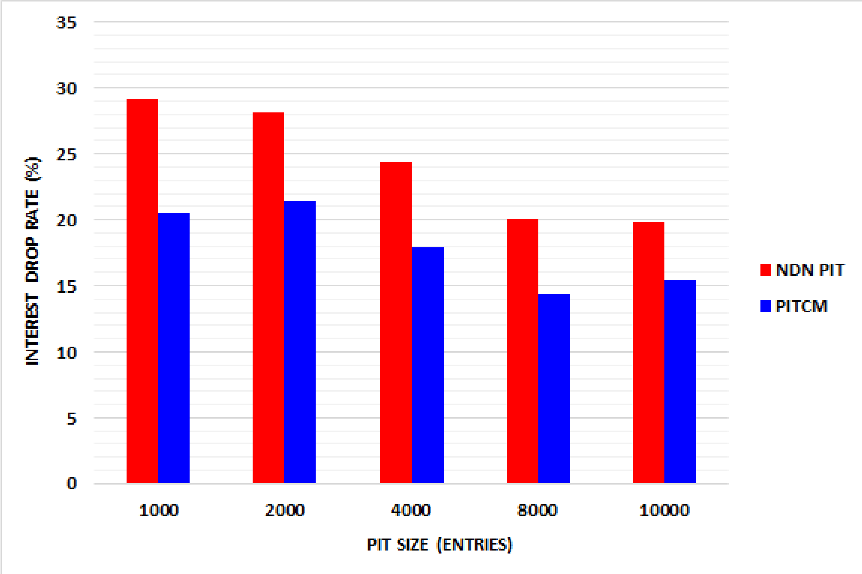


Figure 5.15. Abilene Topology: IDR vs PIT Size for NDN PIT and PITCM

The next Figure 5.15 illustrates the Interest drop rate of PITCM and NDN PIT over different PIT size. As a result, when the PIT size is 2000 entries, a high impact is displayed with a high Interest drop rate, which is somehow (21.4%) in PITCM. In NDN PIT, this percentage increases to (28.09%) at PIT size is 2000 entries, and it gradually decreases as the PIT size increases. However, PITCM still records a better/lower Interest drop rate (i.e., 31.26% as the PIT size is setting 2000 entries) score compared to NDN PIT. NDN decreases the delay by making the NDN routers able to cache the copy from every Data packet pass over NDN routers.

Figure 5.16 presents the performance comparison of Interest satisfaction rate of PITCM and NDN PIT over different PIT size. The Interest satisfaction is (61.97%) with PIT size is 2000 entries in NDN PIT. In PITCM, Interest satisfaction is (74.99%) with PIT size is 2000 entries. Similarly, with the increase of the PIT size to 10000, the Interest satisfaction records (74.63%) in NDN PIT. In PITCM, this reduction is (86.54%). Hence, it guarantees that with the small PIT size scenarios, the Interest satisfaction rate for PITCM can be maximized to (21%) compared to NDN PIT.

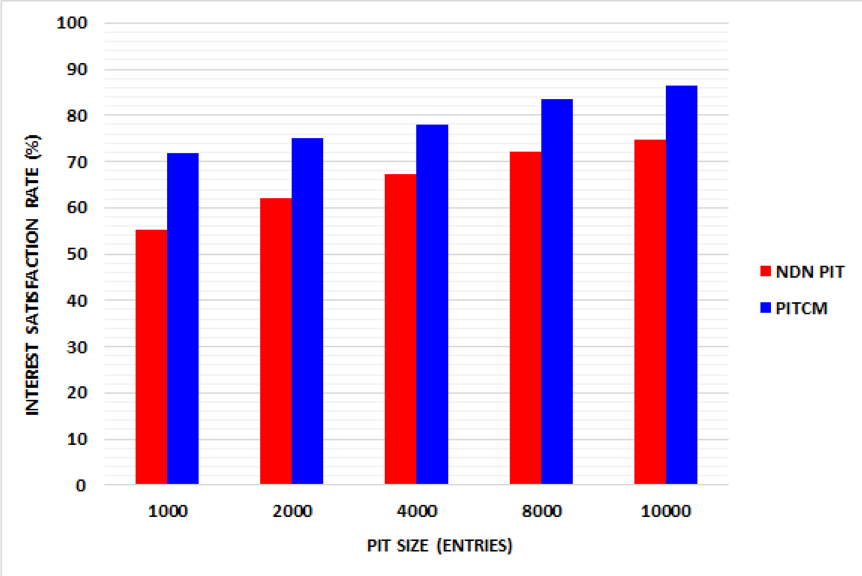


Figure 5.16. Abilene Topology: ISR vs PIT Size for NDN PIT and PITCM

Figure 5.17 shows the Interest satisfaction delay of PITCM and NDN PIT over different PIT size. Both PITCM and NDN PIT presented a gradual decrease of Interest satisfaction delay as the PIT size rises. NDN PIT seems to have a higher Interest satisfaction delay than PITCM. In brief, PITCM achieves between (91.06%) and (95.53%) lower than NDN PIT Interest satisfaction delay when the PIT size is 2000 and 10000 entries, respectively.

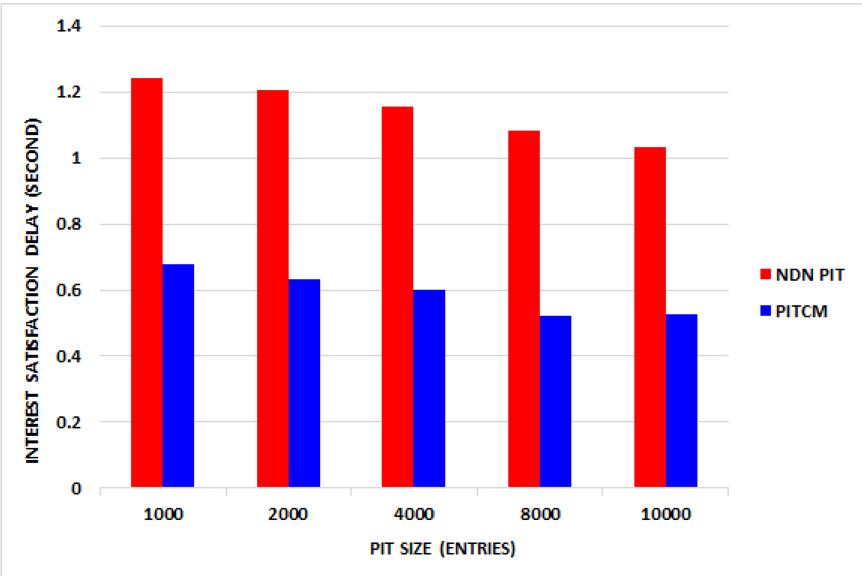


Figure 5.17. Abilene Topology: ISD vs PIT Size for NDN PIT and PITCM

As a conclusion, NDN PIT and PITCM were compared in Abilene topology, and results are shown in Figures 5.13 to 5.17. Based on these figures, it is clear that different PIT size may impact the performance of the PIT since the small size of PIT could cause the PIT overflow in traditional NDN PIT when the number of Interest packet rate increases. On the other hand, when the size of PIT is increased, fairly this may increase the performance of PIT in case of a small network as well as under this work condition. The analysis shows that both NDN PIT and PITCM behave similarly in dealing with various PIT sizes when it is increased from 4000-10000 entries. However, PITCM achieves a better higher Interest satisfaction rate when compared to NDN PIT, and also PITCM reduces significantly the Interest retransmission rate, Interest drop rate and Interest satisfaction delay experienced by the PIT size. In addition, PITCM is able to maintain a PIT length at a small PIT size.

5.2.2.2 Rocketfuel-mapped Scenario

The previous section focused on evaluation PITCM in Abilene topology over different PIT size. This section presents the PITCM evaluation results using Rocketfuel-mapped topology in order to model a more realistic network environment and tested all the expected scenarios.

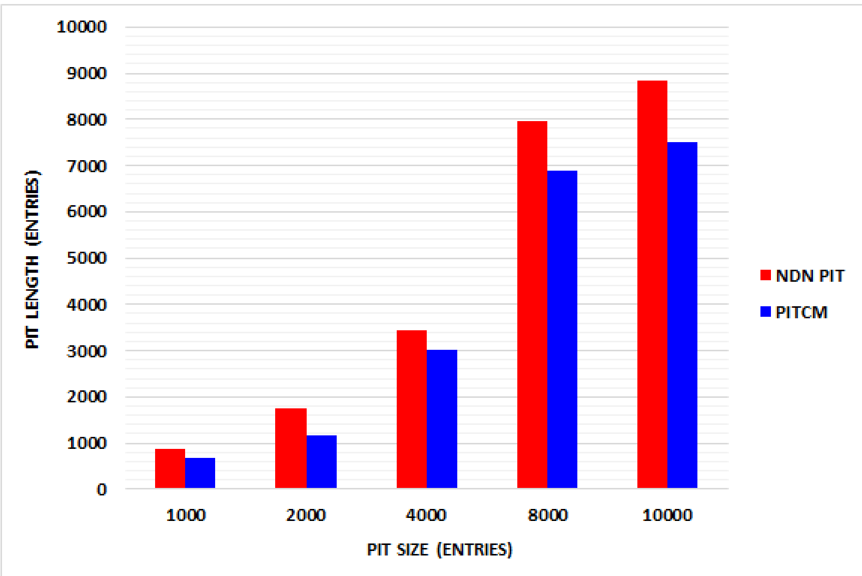


Figure 5.18. Rocketfuel Topology: PIT length vs PIT Size for NDN PIT and PITCM

The result is presented in Figure 5.18, the PIT length of PITCM and NDN PIT over five different sizes of PIT. According to the results obtained, NDN PIT and PITCM were share the same behavior since they are increasing the PIT length as the size of PIT increase which is because the ability of the NDN nodes to cache more of the Interest packet in the PIT. Hence, the simulated PITCM performance is similar, i.e., the difference compared to NDN PIT is decreasing from (51.27% entries) for 2000 entries of PIT size to (17.44% entries) for 10000 entries of PIT size. To conclude, PITCM better control management approach decreases PIT length in order to manage the PIT size, and thus mitigates the occurrence of PIT overflow.

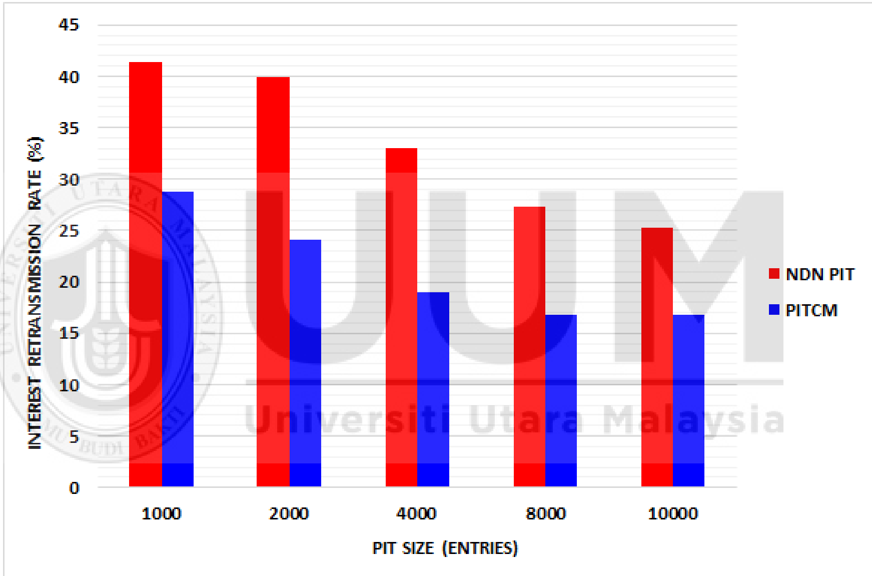


Figure 5.19. Rocketfuel Topology: IRR vs PIT Size for NDN PIT and PITCM

Figure 5.19 displays the Interest retransmission rate of PITCM and NDN PIT under various sizes of PIT. Based on this figure, it could be seen that in both PITCM and NDN PIT, the Interest retransmission rate is (24.153%) and (39.934%), respectively, when the size of PIT is 2000 entries, and the Interest retransmission rate starts decreasing when the size of PIT is increased. One of the main reasons for this is the capacity of PIT. When the capacity of PIT is small, the Interest packet is no longer received; therefore, Interest will be dropped. Consequently, the Interest retransmission will be enlarged. Thus, PITCM achieves (56.364%) and lower than NDN PIT Interest retransmission rate when the size of PIT is 2000 entries. As a result, PITCM faces less Interest retransmission rate as compared to the NDN PIT.

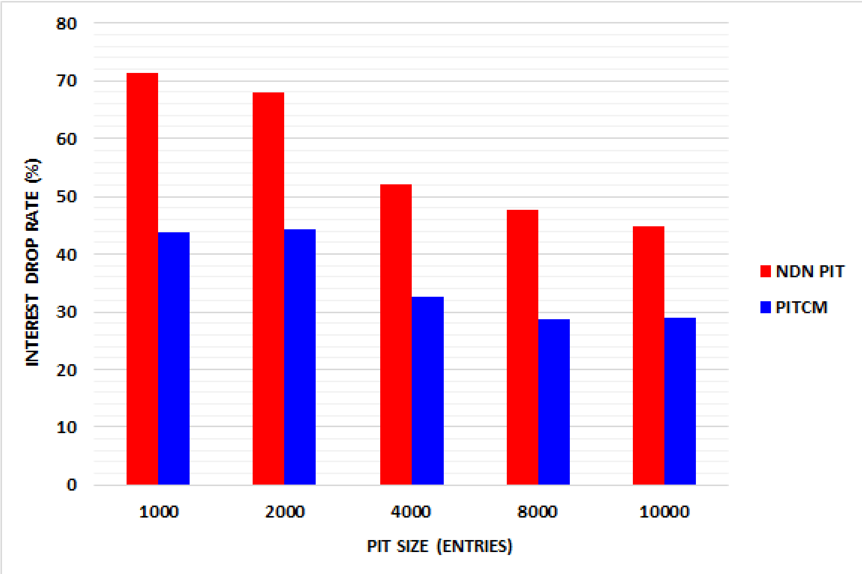


Figure 5.20. Rocketfuel Topology: IDR vs PIT Size for NDN PIT and PITCM

As can be seen in Figure 5.20, PITCM effectively monitors and manages PIT. Consequently, due to PIT overflow, the Interest drop rate in the network increase. Mean of average Interest drop rate in all network routers is (44.19%) when PIT size is 2000 entries for PITCM compared with NDN PIT that increases by (68.04%). Similarly, when the setting of the PIT size is 10000 entries, the Interest drop rate for PITCM records a decrease (29.02%). In NDN PIT, this decrease is (44.79%). Therefore, the PITCM could improve the PIT by (53.97%) when the PIT size is 2000 as compares with NDN PIT.

The Interest satisfaction rate obtained due to different size of PIT is summarized in Figure 5.21. PITCM and NDN PIT, show different Interest satisfaction rate values for different PIT size due to the heavy Interest packet rate. The highest increase in Interest satisfaction with PITCM is (67.45%) for PIT size is 2000 entries, and up to (88.26%) when PIT size is setting to 10000 entries. NDN PIT is obtained (32.35%) and (61.02%) Interest satisfaction rate for PIT size is 1000 and 10000 entries, respectively. As a

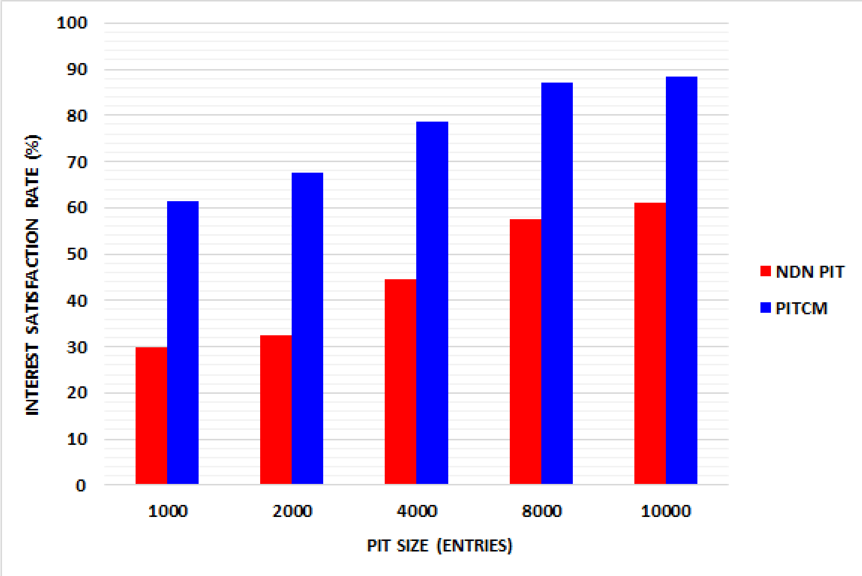


Figure 5.21. Rocketfuel Topology: ISD vs PIT Size for NDN PIT and PITCM

result, PITCM faces highest Interest satisfaction during the content retrieval process as compared to the NDN PIT.

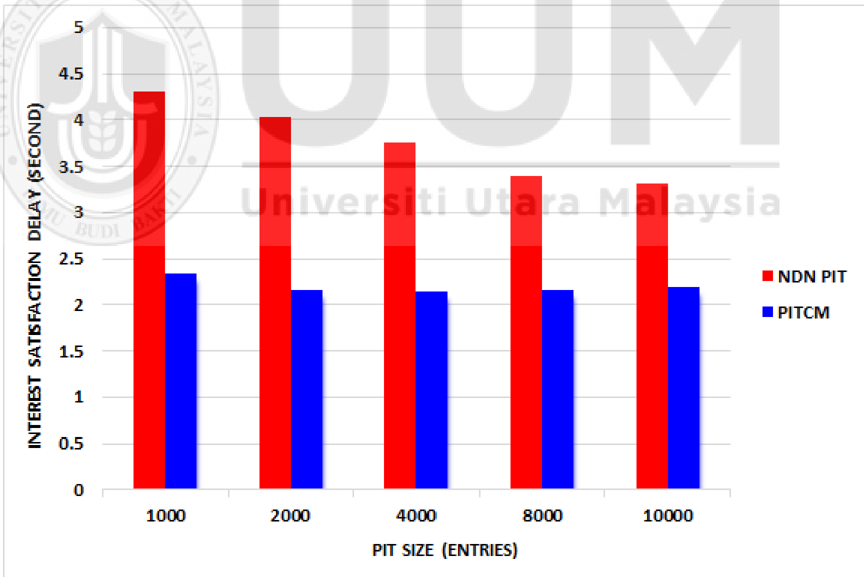


Figure 5.22. Rocketfuel Topology: ISD vs PIT Size for NDN PIT and PITCM

Figure 5.22 illustrates the Interest satisfaction delay of PITCM and NDN PIT over five different PIT sizes. Over all results for Interest satisfaction delay, the performance of PITCM decreases and depends on the PIT size. However, performance of NDN PIT degrades due to the increase in the size of PIT. The performance of PITCM decreases and remains constant as the PIT size increases. Thus, PITCM achieves between (86.22%) and (50.98%) lower than NDN PIT Interest satisfaction delay, when the PIT size is 2000 and 10000 entries, respectively. Overall, PITCM faces less delay during the content retrieval process compared to the NDN PIT.

Overall, PITCM and NDN PIT were compared by using Rocketfuel-mapped topology and the results are shown in Figures 5.18 to 5.22. When the network is large and the number of Interests is also large, the PIT size will grow rapidly. Increasing the PIT size may lead to caching more entries; nevertheless, it can be too large to be stored in a cache and to be processed quickly for looking up or updating entries as argued in [51]. Hence, simulation results indicate the effectiveness of this approach in both reducing PIT length, Interest satisfaction delay, Interest retransmission rate, Interest drop rate, and increasing Interest satisfaction rate for subscribers. It is therefore possible to argue that the PITCM provides optimizing for the performance of PIT for a given network scenario.

5.3 Discussion on PITCM Performance

The previous sections presented the analyses of the PITCM against the NDN PIT. This section discusses the PITCM's performance in NDN based on the experimental results obtained in Section 5.2. This section provides a summary of several tests conducted to look into the behavior of the PITCM under a variety of PIT size and network conditions. Specifically, this section illustrates the effects of each individual factor, namely the PIT size, the number of Interests, the network topology, the performance metrics, as well as a combination of these factors on the PITCM's performance, which are as follows:

- Impact of various Interest rates: the low or high Interest rates over the network may affect the performance of PIT based on the PIT conditions. Based on

Figures 5.3 to 5.12, it is observed that, as a general trend, the performance of PITCM and NDN PIT would increase as the Interest packet rate increases. However, the Interest drop, Interest retransmission and Interest satisfaction delay in the PITCM have dramatically decreased as compared to that in the NDN PIT. On the other hand, the Interest satisfaction is increased in PITCM as compared to that in the NDN PIT. Thus, PITCM's quick response to the Interest packet rate changes based on the it is behavior with PIT as compared to NDN PIT which has poor performance when the Interest packet rate is increased.

- Impact of various PIT sizes: when the PIT size is large, PIT is capable to catch amount of entries as much as possible. Hence, the performance of PIT depends on the behavior of the network and PIT condition. Based on Figures 5.13-5.22, it was observed that, as a general trend, the performance of PITCM and NDN PIT would increase as the PIT size increases. However, the Interest drop, Interest retransmission and Interest satisfaction delay in the PITCM have decreased better than the standard NDN PIT, during comparing. On the other hand, the interest satisfaction is raised in PITCM as compared to that in the NDN PIT.
- Impact of different topologies: the Named Data Network could be represented in a variety of topologies depending on the implementation environments. However, the topologies are a critical issue because they have a direct impact on the results of simulation [319]. Among the most popular static topologies are Abilene and Rocketfuel-mapped AT&T. The PITCM was evaluated with 46 nodes of Abilene and 625 nodes of Rocketfuel-mapped AT&T topology. In these topologies, the PITCM has achieved a better performance as compared to the NDN PIT.
- Impact of performance metrics: choosing relevant metrics is very important in order to ensure the performance of PITCM. Different metrics, such as PIT length, Interest drop, Interest satisfaction delay, Interest retransmission, and In-

terest satisfaction have been determined for evaluating and analyzing in this study. Among these metrics, Interest drop Interest retransmission, and Interest delay have the most significant negative impact on the PIT performance, which happens after the PIT overflows. In addition, adapting the PIT length is affecting not only the performance of the PIT but also the whole network.

5.4 Summary

PITCM was developed to extend the PIT functionalities for NDN. PITCM was designed and implemented by using ndnSIM simulator environment. Following this, the PITCM was evaluated by observing different performance metrics over a variety of scenarios. Performance of the overall system is measured in term of PIT length, Interest satisfaction, Interest drop, Interest retransmission and Interest satisfaction delay as compared to that of NDN PIT in different topologies. Thus, the obtained results showed that PITCM outperforms with NDN PIT. As a result, PITCM has proved to be the proper choice for developers and normal subscribers in NDN.

CHAPTER SIX
CONCLUSION AND FUTURE WORK

The time when the Internet was designed, i.e., late 1960s, it has played a more important part in people's life. From the earliest starting point, the Internet keeps running on top of the protocol stack of the Transmission Control Protocol/Internet Protocol (TCP/IP) with the aim of connection a few of machines. The existing Internet architecture is known as a host-centric architecture. The entire Internet data transmission is acknowledged by building up the correspondence channels. This host-driven Internet has superbly coordinated the early Internet use, which is not complicated. Essentially, the Internet applications or protocols are all host-to-host correspondences, such as instant chat messages, web surfing, email forwarding or File Transfer Protocol (FTP) file downloading [25].

However, major architectural techniques and theoretical updates have been productive without dissociating their original semantics. The initial Internet design was a model to share little memory requiring resources when compared to the needs of its users at present. The requires of current Internet usage have boosted the usability to the staggering traffic and memory growth in the capacity of the data traversed via user connectivity and host-to-host interactivity [220]. These along with the predicted traffic gave rise to the leading research of Named Data Network (NDN). The NDN architecture that yearns to bring information which is much closer to the subscribers by dissociating partly the address of the host in lieu of the content names. NDN has been agreed as a future paradigm of in-network (especially in PIT) form of communication.

This thesis addressed the issue of flow PIT sizing of the forwarding system and how this issue can affect the performance of PIT in NDN. The previous chapters proposed a new approach, namely Pending Interest Table Control Management (PITCM) approach that consists of Adaptive Virtual Pending Interest Table (AVPIT) mechanism, Smart Threshold Interest Lifetime (STIL) mechanism, and Highest Lifetime Least Request (HLLR) policy. This approach has been implemented and performance analysis based on the numeric results obtained from simulation. Chapter Six provides summaries of the thesis research in Section 6.1. Section 6.2 presents the research limitations. Contributions of this thesis are illustrated in Section 6.3. Finally, Section 6.4 provides some recommendations for future studies.

6.1 Summary of the Thesis

In spite of the numerous advantages of NDN router tables, several hardware challenges are faced, which include the PIT in terms of Interest packets processing in the NDN. In addition, the PIT size that requires to be control and monitored incoming Interest packet as well as to be managed the entries inside the PIT. The problem becomes more critical when the impending PIT overflows during the heavy load of Interest packets. Therefore, the main objective of this thesis was to design and implement the Pending Interest Table Control Management (PITCM) approach in order to address the PIT overflow due to the increased Interest arrival rate to the NDN router, which can affect the entire network.

As mentioned in Chapter One, the aim of this research was to highlight the problems faced by PIT in the NDN environment and to introduce the issues required to be addressed. In addition, Chapter One explained the main objectives and the significance of this work. Continuously, the brief overview of the NDN architecture was provided as a future Internet paradigm. Next, the PIT overflow problem was presented which showed that the PIT is affected. This leads to increased delay, retransmission and packet drop that affects the utilization of PIT and the overall network performance.

It was concluded that the current PIT cannot handle the challenge caused by the increased Interest packet rate with high Interest lifetime that needs to be stored in the PIT. In addition, limited PIT size leads the NDN router to be overflowed rapidly, thus resulting in a negative impact on the PIT performance. In addition, the study specifies a technical background of the research work by reviewing the core properties of the NDN architecture that are essential to describe this work. Moreover, this chapter presents brief descriptions of the principles and operations of the PIT regarding the PIT management techniques, as discussed in detail in Chapter Two.

A specific framework was introduced as guidelines that were used to accomplish this research. In order to achieve these objectives, several methods have been used for the implementation of this research that focuses on the management of PIT, which is taken by the researcher in order to answer all questions of the research to tally out with the objectives of the study.

Chapter Four elaborates on the design of the proposed components of the approach. Firstly, design an AVPIT mechanism that predicts the impending overflow of the PIT. The queuing theory is the basis of the AVPIT design since the Adaptive Virtual Queue concept was used as a part of this mechanism. Simulation results showed that AVPIT improves PIT overflow prediction and reaction. Second, design an SITL mechanism that adjusts the Interest packet lifetime in order to avoid overstaying of PIT entries during heavy network load. STIL mechanism was inspired by a renewal theory in order to adapt Interest packet lifetime only to Interest packet that belongs to the face which has a heavy Interest packet received. STIL adapted Interest packet lifetime for new incoming Interest packet during the treatment of the impact of VPIT overflow. Simulation experiments had confirmed that the proposed mechanism controls the impact of the highest Interest packet lifetime on the performance of PIT without increasing unnecessary Interest retransmission. This proposed mechanism was validated using simulation experiments.

Next, design an HLLR replacement policy that determines PIT entry to be replaced for efficient PIT utilization. Since replacing the entry is not an easy decision, the victim packet should be selected carefully. Scheduling theory was a basis to determine the unpopular entry that has the longest lifetime with a minimum frequency request inside a PIT. Simulation experiments had confirmed that the proposed policy support the management of PIT entries by using lifetime and Interest frequency factors to determine the replacement entry. This proposed policy was validated using simulation experiments.

The combination of the AVPIT, STIL and HLLR in PITCM approach is a distinctive aspect of this research. Hence, Chapter Five presents an evaluation of the performance of the approach. This evaluation is achieved by means of the results obtained from the simulation experiments of the designed approach and the comparisons with traditional NDN PIT. The results demonstrated that our approach improves the PIT length, the Interest retransmission rate, the Interest drop rate, the Interest satisfaction delay and the Interest satisfaction rate compared with the current works (i.e., traditional NDN PIT).

The findings have significant implications represented by providing a reliable approach within NDN architecture. By conducting the PITCM performance evaluation, the results certainly emphasize the fact that the framed objectives of this research have been completely achieved.

6.2 Contributions of the Thesis

The overall contribution of this research was to design and implement a Adaptive Virtual Pending Interest Table (AVPIT) mechanism, Smart Threshold Interest Lifetime (STIL) mechanism, Highest Lifetime Least Request (HLLR) policy and combination them in Pending Interest Table Control Management (PITCM) approach in order to enhance the management operations of PIT. PITCM has ability for monitoring and controlling the incoming Interest packets as well as managing the PIT entries, which mitigate the PIT overflow. PITCM addresses the relationship between the PIT size and its overflow in order to enhance utilization of PIT in NDN routers. This increasing can mitigate the Interest drop, Interest delay, unnecessary Interest retransmission as well. The contributions of the research cover a variety of aspects as follows:

i. Designing and implementing an adaptive Virtual Pending Interest Table (AVPIT) mechanism for improvement accuracy of obtaining early PIT overflow prediction and reaction. The AVPIT consists of three components that are, namely overflow indicator, operation and feedback methods.

   a. An analytical model for AVPIT mechanism based on Actives Queue Management concept to study the impact of increasing the overload and PIT size conditions on the PIT overflow.

   b. The AVPIT mechanism from analytical model equations was transformed in to a programmable code using C/C++ Development Tool (CDT) that runs on top of the Eclipse platform.

   c. Graphical comparison technique was used to validate the AVPIT mechanism, which was achieved by comparing the model results with the results obtained from a valid network simulator. For this purpose, a Dumbbell topology is considered with a variety of scenarios using the ndnSIM simulator. AVPIT was validated to ensure that it meets the intended requirements in terms of PIT length, and Interests drop based on NDN router.

d. The verification and evaluation of the proposed AVPIT were performed by implementing it in ndnSIM simulation environment and comparing the identified network state by NDN based-AVPIT and the traditional NDN (NDN PIT) environment. Tree topology is considered with a variety of scenarios regards of PIT length, and Interests packet drop based on NDN router. AVPIT evaluation was focused on considering the relationship between the PIT overflow and the number of PIT entries.

e. AVPIT significantly impacted on PIT performance via reducing Interest packet drop and waiting time for subscribers nodes by informing them faster than Interest packet timeout.

ii. Designing and implementing Smart Threshold Interest Lifetime (STIL) mechanism to avoid overstaying of PIT entries during heavy network load and enhance PIT performance.

a. An analytical model for STIL mechanism to make NDN router grants higher/shorter lifetime value for incoming Interest packet. Renewal theory features and properties were carried out, which is able to accurately adapt Interest packet lifetime according to the AVPIT and network load conditions.

b. The STIL mechanism from analytical model equations was transformed in to a programmable code using C/C++ Development Tool (CDT) that runs on top of the Eclipse platform.

c. Performance testing technique was used to validate the STIL mechanism, which was used for testing whether all characteristics of the performance are measured as well as evaluated with sufficient accuracy. To validate the accuracy of the proposed STIL, Tree topology is considered with a variety of scenarios using the ndnSIM simulator. Thus, the validation of STIL focused on examining the relationship between PIT overflow and the lifetime value of Interest packet over two different ways, including a numerical analysis, and simulation scenarios.

d. The verification and evaluation of the proposed STIL were obtained by implementing it in ndnSIM and evaluating by comparing the obtained results with DPEL mechanism and the traditional NDN PIT. Abilene topology was used to evaluate STIL regards two types of performance metrics, which are Interest packet retransmission and Interest stratification rate based on NDN router.

e. STIL significantly impacted on PIT performance without increasing Interest packet retransmission, which results in a better PIT management.

iii. Designing and implementing Highest Lifetime Least Request (HLLR) policy of PIT to assist the management of PIT entries, which is applicable for PIT since entry lifetime and incoming face factors are already present in the PIT concepts.

a. An analytical model for HLLR policy based on Scheduling theory features. It is formulated a general rule that is a function of the entry lifetime and frequency of entry that yield an efficient priority for replacement when indicate the VPIT is overflowed.

b. The HLLR policy from analytical model equations was transformed in to a programmable code using C/C++ Development Tool (CDT) that runs on top of the Eclipse platform.

c. Sensitivity analysis technique was used to validate the HLLR policy, which was performed by methodically changing the input values and parameters of the proposed model across a range of interests and taking into consideration the effect on the behavior of the model. For validating HLLR policy, two methods were employed, firstly by numerical analysis via executing one operation in order to ascertain how the HLLR policy works. The second method to validate the accuracy of the HLLR policy was by means of the simulation scenario over Tree topology regards Interest retransmission rate.

d. The verification and evaluation of the proposed HLLR were obtained by implementing it in ndnSIM. In order to guarantee the consistency of the presented results, the scenarios were tested HLLR using the Abilene topology and evaluating it by comparing the obtained results with Persistent, Random and LRU policies.

e. HLLR significantly impacted on PIT performance via minimizing searching time during content eviction and increase PIT utilization as well.

iv. Developing and evaluating PITCM approach in NDN environment.

a. Incorporation of the proposed AVPIT, STIL and HLLR in PITCM was achieved.

b. The evaluation of PITCM over a variety of scenarios was illustrated by implementing it in ndnSIM. PITCM and conventional NDN PIT were test under different packet rate (i.e., 1000-10000 Interest/second)), and under different PIT size (i.e., 1000-10000 entries) over Abilene topology and over a modified version of a Rocketfuel topology. PITCM was evaluated by observing different performance metrics over a variety of scenarios. Performance of the overall system is measured in term of PIT length, Interest satisfaction, Interest drop, Interest retransmission and Interest satisfaction delay, which is used to measure the performance of the PITCM.

c. The comparison of PITCM with traditional NDN PIT was performed to show the significant performance that was obtained. Thus, the obtained results showed that PITCM outperforms with NDN PIT over NDN architecture.

6.3 Research Limitation

Although this research was conducted under careful selection and procedure including the conceptual model, implementation, verification, validation and evaluation, it is limited to specific and precise usage. First of all, this work was conducted in specific network topologies used in the implementation that are widely used and proved in Named Data Networks. However, this work did not include all the available topologies. In addition, the number of nodes used in the validation and performance evaluation was limited and fixed, whereas in real NDN, it is unpredictable and changeable. Moreover, link failure and other losses are not discussed since it presumably would not affect the result.

6.4 Recommendation for Future Work

Finally, some topics can be suggested for future work:

i. Performance evaluation of the PITCM approach in a testbed is another way to extend this research. Although the PITCM was evaluated comprehensively and extensively through a validated simulator, its implementation in a real testbed is definitely of a great interest. However, due to the nature of NDN, this is a future architecture and is not deployed yet, and it is quite complicated to evaluate the performance on the testbed. Furthermore, to evaluate the PITCM using real traffic is surely a good extension to be performed in extending the scope of this research.

ii. For the PITCM proposal in reporting the performance of Pending Interest Table, it is important to examine more ICN tables, such as Content Store (CS) as it also can become bottleneck [233] that affect the performance of the system when the number of Data packets are increased over time. The same concept of PITCM can be applied on the CS in order to predict and mitigate the CS overflow.

iii. The proposed PIT is intended for the most pessimistic scenario of flow control, yet it would not have the capacity to handle an Interest flooding attack, where each Interest packet remains for Interest lifetime expire. For this situation, measurements gathered on the NDN routers ought to have the capacity to distinguish the attacks, and afterward apply countermeasures. The attacker may likewise choose expansive qualities for the Interest packet lifetime field. Thus, the proposed approach is utilize and alter to have the capacity to redesign the lifetime field under system control when the PIT overflows. In such a way, after making modification, the proposed approach could possibly address the Interest flooding issue and upgrade the lifetime field under system control when the PIT overflows. This is another way to extend scalable PIT approach in the future work.

iv. One more instance for the future work is that the NDN approach can be potentially deployed in a Mobile Ad-hoc Network (MANET) environment. Meisel *et al.* in [138] were the first to argue that MANET can be made more effective and efficient through NDN. However, MANET needs modifications to effectively cope with this kind of network, where the proactive Interests packet overflow announcement populate the PITs, which is unacceptable in high mobility MANET because of the huge control overhead. In addition, the broadcast wireless channel enabled packet overhearing, thus helping to identify neighbors and to cache data, but it may cause stable/scalable problems. For this situation, utilizing PITCM approach in the MANET can accomplish better results.

v. In the original design, the NDN router applies a centralized PIT since the PIT is located and maintained in a single location inside NDN routers. Under certain conditions, a centralized PIT can be beneficial, and it may get a better perspective on the big picture of the network, particularly in a small network. Hence, it is easier to be used by the NDN router due to the simplicity of having a single PIT design. Moreover, Interest packet kept in the same location is easy to be inserted, updated, managed, monitor or analyzed. However, the centralized PIT also has certain limitations when the network becomes large, including a higher initial setup cost, a single point of failure or PIT overflow that could bring down the entire network, and requires to have large storage with higher processing capabilities. In order to meet these challenges, decentralization is the key to disperse the Interest packet decision-making authority in an NDN routers to popular and unpopular.

REFERENCES

[1] G. M. Brito, P. B. Velloso, and I. M. Moraes, *Information Centric Networks: A New Paradigm for the Internet*. John Wiley and Sons, 2013.

[2] W. Li, S. M. Oteafy, and H. S. Hassanein, "Dynamic Adaptive Streaming over Popularity-driven Caching in Information-Centric Networks," in *2015 IEEE International Conference on Communications (ICC)*. IEEE, 2015, pp. 5747–5752.

[3] K. Michael, "Growth of the Internet 2014," Internet Society Organization, Tech. Rep., 2014, accessed: 15-Jun-2014. [Online]. Available: https://www.internetsociety.org/sites/default/files/Global_Internet_Report_2014.pdf

[4] ICNRG-IRTF, "Information-Centric Networking Research Group," January 2016, accessed: 15-Jul-2013. [Online]. Available: https://irtf.org/icnrg

[5] M. Hovaidi Ardestani, "Congestion Control in Information Centric Networking using Neural Networks," Ph.D. dissertation, Aalto University, 2014.

[6] J. Scott, *Social Network Analysis*. Sage, 2012.

[7] K. Hampton, L. S. Goulet, L. Rainie, and K. Purcell, "Social Petworking Sites and Our Lives," *Pew Research Center's Internet and Amercan Life Project*, pp. 1–85, 2011, accessed: 17-May-2013. [Online]. Available: http://cn.cnstudiodev.com/uploads/document_attachment/attachment/46/pew_-_social_networking_sites_and_our_lives.pdf

[8] K. Yuta, N. Ono, and Y. Fujiwara, "A Gap in the Community-Size Distribution of a Large-Scale Social Networking Site," *arXiv preprint physics*, pp. 1–9, 2007, accessed: 10-Jul-2013. [Online]. Available: https://arxiv.org/pdf/physics/0701168.pdf

[9] P. Juluri, V. Tamarapalli, and D. Medhi, "Measurement of Quality of Experience of Video-on-Demand Services: A Survey," *IEEE Communications Surveys & Tutorials*, vol. 18, no. 1, pp. 401–418, 2016.

[10] D. Ciullo, V. Martina, M. Garetto, and E. Leonardi, "How Much can Large-scale Video-on-Demand Benefit from Users' Cooperation?" *IEEE/ACM Transactions on Networking*, vol. 23, no. 6, pp. 1846–1861, 2015.

[11] M. Amadeo, C. Campolo, A. Molinaro, and N. Mitton, "Named Data Networking: a Natural Design for Data Collection in Wireless Sensor Networks," in *Proceedings of Wireless Days (WD) - 2013 IFIP*. IEEE, 2013, pp. 1–6.

[12] S. Devaraj, D. Ortiz, S. Chinnasamy, and R. MELENDRES, "System and Method for On-line Multi-player Interactive Wagering," accessed: 9-Apr-2014. [Online]. Available: https://www.google.com/patents/US20160300432

[13] M. W. Yacenda, "Interactive Computer Gaming System with Audio Response," accessed: 18-Apr-2014. [Online]. Available: https://www.google.com/patents/US20010003100

[14] M. Amadeo, C. Campolo, and A. Molinaro, "Internet of Things via Named Data Networking: the Support of Push Traffic," in *Proceedings of 2014 International Conference and Workshop on the Network of the Future (NOF)*. IEEE, 2014, pp. 1–5.

[15] M. Loffler and A. Tschiesner, "The Internet of Things and the Future of Manufacturing," *McKinsey on Business Technology*, vol. 30, pp. 8–13, 2013.

[16] L. Atzori, A. Iera, and G. Morabito, "The Internet of Things: A Survey," *Computer networks*, vol. 54, no. 15, pp. 2787–2805, 2010.

[17] M. Gallo, "Traffic and Resource Management in Content-Centric Networks: Design and Evaluation," Ph.D. dissertation, TELECOM ParisTech, 2012.

[18] D. Papadimitriou, H. Tschofenig, A. Rosas, S. Zahariadis *et al.*, "Fundamental Limitations of current Internet and the path to Future Internet," *European Commission FIArch Group*, vol. 1, pp. 1–15, 2011.

[19] GENI, "The Global Environment for Network Innovations," accessed: 24-Aug-2014. [Online]. Available: http://www.geni.net

[20] FIA, "European Future Internet Assembly," accessed: 5-Jun-201. [Online]. Available: http://www.future-internet.eu

[21] AsiaFI, "Asia Future Internet Forum," accessed: 12-Feb-2013. [Online]. Available: http://www.asiafi.net/

[22] D. P. Arjunwadkar, "Introduction of NDN with Comparison to Current Internet Architecture based on TCP/IP," *International Journal of Computer Applications*, vol. 105, no. 5, pp. 31–35, 2014.

[23] T. Zahariadis, D. Papadimitriou, H. Tschofenig, S. Haller, P. Daras, G. D. Stamoulis, and M. Hauswirth, "Towards a Future Internet Architecture Theodore," in *The Future Internet Assembly*. Springer, 2011, pp. 7–18.

[24] Cisco, "Cisco Visual Networking Index:Global Mobile Data Traf- fic Forecast Update, 2014-2019," Cisco, Technical Report, Tech. Rep., 2015.

[25] A. V. Vasilakos, Z. Li, G. Simon, and W. You, "Information Centric Network: Research Challenges and Opportunities," *Journal of Network and Computer Applications*, vol. 52, pp. 1–10, 2015.

[26] F. L. F. Almeida and J. M. Lourenco, "Information Centric Networks-Design Issues, Principles and Approaches," *International Journal of Latest Trends in Computing*, vol. 3, no. 3, pp. 58–66, 2012.

[27] B. Mathieu, P. Truong, J.-F. Peltier, Y. Wei, and G. Simon, *Media Networks - Architectures, Applications, and Standards*. CRC Press, 2012, ch. Information-Centric Networking: Current Research Activities and Challenges, pp. 141–184.

[28] A. Ghodsi, S. Shenker, T. Koponen, A. Singla, B. Raghavan, and J. Wilcox, "Information-Centric Networking: Seeing the Forest for the Trees," in *Proceedings of the 10th ACM Workshop on Hot Topics in Networks*. ACM, 2011, pp. 1–6.

[29] N. Fotiou, P. Nikander, D. Trossen, and G. C. Polyzos, "Developing Information Networking Further: From PSIRP to PURSUIT," in *Proceedings of International Conference on Broadband Communications, Networks and Systems*. Springer, 2010, pp. 1–13.

[30] D. Kutscher, H. Flinck, and H. Karl, "Information-Centric Networking: A position paper," in *Proceedings of 6th GI/ITG KuVS Workshop on Future Internet*, 2010, pp. 1–2.

[31] Z. Jaffri, Z. Ahmad, and M. Tahir, "Named Data Networking (NDN), New Approach to Future Internet Architecture Design: A Survey," *International Journal of Informatics and Communication Technology (IJ-ICT)*, vol. 2, no. 3, pp. 155–165, 2013.

[32] T. Koponen, M. Chawla, B.-G. Chun, A. Ermolinskiy, K. H. Kim, S. Shenker, and I. Stoica, "A Data-Oriented (and Beyond) Network Architecture," *ACM SIGCOMM Computer Communication Review*, vol. 37, no. 4, pp. 181–192, 2007.

[33] C. Dannewitz, "NetInf: An Information-Centric Design for the Future Internet," in *Proceedings of 3rd GI/ITG KuVS Workshop on The Future Internet*, 2009, pp. 1–3.

[34] V. Jacobson, D. K. Smetters, J. D. Thornton, M. F. Plass, N. H. Briggs, and R. L. Braynard, "Networking Named Content," in *Proceedings of the 5th International Conference on Emerging Networking Experiments and Technologies*. ACM, 2009, pp. 1–12.

[35] L. Zhang, D. Estrin, J. Burke, V. Jacobson, J. D. Thornton, D. K. Smetters, B. Zhang, G. Tsudik, D. Massey, C. Papadopoulos *et al.*, "Named Data Networking (NDN) Project," PARC Technical Report 2010-003, Tech. Rep., 2010. [Online]. Available: http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.366.6736&rep=rep1&type=pdf

[36] M. Tortelli, G. Piro, L. A. Grieco, and G. Boggia, "On Simulating Bloom Filters in the ndnSIM Open Source Simulator," *Simulation Modelling Practice and Theory*, vol. 52, pp. 149–163, 2015.

[37] V. Jacobson, J. Burke, D. Estrin, L. Zhang, B. Zhang, G. Tsudik, K. Claffy, D. Krioukov, D. Massey, C. Papadopoulos *et al.*, "Named Data Networking (NDN) Project 2012-2013 Annual Report," Tech. Rep., 2014.

[38] D. Saxena, V. Raychoudhury, N. Suri, C. Becker, and J. Cao, "Named Data Networking: A Survey," *Computer Science Review*, vol. 19, pp. 15–55, 2016.

[39] M. Amadeo, C. Campolo, and A. Molinaro, "Forwarding Strategies in Named Data Wireless Ad hoc Networks: Design and Evaluation," *Journal of Network and Computer Applications*, vol. 50, pp. 148–158, 2015.

[40] C.-Y. Ho, C.-Y. Ho, and C.-C. Tseng, "A Case Study of Cache Performance in ICN - Various Combinations of Transmission Behavior and Cache Replacement Mechanism," in *Proceedings of 2015 17th International Conference on Advanced Communication Technology (ICACT)*. IEEE, 2015, pp. 323–328.

[41] H. Moustafa and S. Zeadally, *Media Networks: Architectures, Applications, and Standards*. CRC Press, 2012.

[42] M. M. S. Soniya and K. Kumar, "A Survey on Named Data Networking," in *Proceedings of 2015 2nd International Conference on Electronics and Communication Systems (ICECS)*. IEEE, 2015, pp. 1515–1519.

[43] F. Oehlmann, "Content-Centric Networking," *Network*, vol. 43, pp. 11–18, 2013.

[44] Z. Zhu, A. Afanasyev, and L. Zhang, "A New Perspective on Mobility Support," Named-Data Networking Project,Techncal Report, Tech. Rep., 2013, accessed: 23-Jan-2015. [Online]. Available: https://named-data.net/wp-content/uploads/TRmobility.pdf

[45] S. DiBenedetto, C. Papadopoulos, and D. Massey, "Routing Policies in Named Data Networking," in *Proceedings of the ACM SIGCOMM Workshop on Information-Centric Networking*. ACM, 2011, pp. 38–43.

[46] H. Dai, B. Liu, Y. Chen, and Y. Wang, "On Pending Interest Table in Named Data Networking," in *Proceedings of the Eighth ACM/IEEE Symposium on Architectures for Networking and Communications Systems*. ACM, 2012, pp. 211–222.

[47] A. Afanasyev, I. Moiseenko, L. Zhang *et al.*, "ndnSIM: NDN Simulator for NS-3," University of California, Los Angeles, Technical Report NDN-0005, Tech. Rep., 2012, accessed: 29-June-2013. [Online]. Available: https://named-data.net/wp-content/uploads/TRndnsim.pdf

[48] C. Tsilopoulos, G. Xylomenos, and Y. Thomas, "Reducing Forwarding State in Content-Centric Networks with Semi-Stateless Forwarding," in *Proceedings of IEEE INFOCOM 2014-IEEE Conference on Computer Communications*. IEEE, 2014, pp. 2067–2075.

[49] A. Azgin, R. Ravindran, and G. Wang, "Mobility Study for Named Data Networking in Wireless Access Networks," in *Proceedings of 2014 IEEE International Conference on Communications (ICC)*. IEEE, 2014, pp. 3252–3257.

[50] G. Carofiglio, M. Gallo, L. Muscariello, and D. Perino, "Pending Interest Table Sizing in Named Data Networking," in *Proceedings of the 2nd International Conference on Information-Centric Networking*. ACM, 2015, pp. 49–58.

[51] C. Park, T. Kwon, and Y. Choi, "Scalability Problem for Interest Diffusion in Content-Centric Network," in *Proceedings of 14th Conference on Next Generation Communication Software (NCS)*, 2010, pp. 58–66.

[52] H. Yuan and P. Crowley, "Scalable Pending Interest Table Design: From Principles to Practice," in *Proceedings of IEEE INFOCOM 2014-IEEE Conference on Computer Communications*. IEEE, 2014, pp. 2049–2057.

[53] Z. Zhou, X. Tan, H. Li, Z. Zhao, and D. Ma, "MobiNDN: A Mobility Support Architecture for NDN," in *Proceedings of 2014 33rd Chinese Control Conference (CCC)*. IEEE, 2014, pp. 5515–5520.

[54] Z. Li, J. Bi, S. Wang, and X. Jiang, "Compression of Pending Interest Table with Bloom Filter in Content Centric Network," in *CFI '12: Proceedings of the 7th International Conference on Future Internet Technologies*. ACM, 2012, pp. 1–4.

[55] M. Varvello, D. Perino, and L. Linguaglossa, "On the Design and Implementation of a Wire-speed Pending Interest Table," in *Proceedings of 2013 IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS)*. IEEE, 2013, pp. 369–374.

[56] W. So, A. Narayanan, and D. Oran, "Named Data Networking on a Router: Fast and DoS-resistant Forwarding with Hash Tables," in *Proceedings of the Ninth ACM/IEEE Symposium on Architectures for Networking and Communications Systems*. IEEE Press, 2013, pp. 215–226.

[57] P. Gasti, G. Tsudik, E. Uzun, and L. Zhang, "DoS and DDoS in Named Data Networking," in *Proceedings of 2013 22nd International Conference on Computer Communication and Networks (ICCCN)*. IEEE, 2013, pp. 1–7.

[58] M. Virgilio, G. Marchetto, and R. Sisto, "PIT Overload Analysis in Content Centric Networks," in *Proceedings of the 3rd ACM SIGCOMM Workshop on Information-Centric Networking*. ACM, 2013, pp. 67–72.

[59] A. J. Abu, B. Bensaou, and A. M. Abdelmoniem, "Inferring and Controlling Congestion in CCN via the Pending Interest Table Occupancy," in *Local Computer Networks (LCN), 2016 IEEE 41st Conference on*. IEEE, 2016, pp. 433–441.

[60] M. Vahlenkamp, "Threats on Information-Centric Networking," Master's thesis, Faculty of Engineering and Computer Science, 2013, accessed: 23-Feb.-2014. [Online]. Available: https://www.inet.haw-hamburg.de/teaching/ws-2012-13/master-projekt/markus-vahlenkamp_seminar.pdf

[61] S. H. Bouk, S. H. Ahmed, M. A. Yaqub, D. Kim, and M. Gerla, "DPEL: Dynamic PIT Entry Lifetime in Vehicular Named Data Networks," *IEEE Communications Letters*, vol. 20, no. 2, pp. 336–339, 2016.

[62] L. Wang, R. Wakikawa, R. Kuntz, R. Vuyyuru, and L. Zhang, "Data Naming in Vehicle-to-Vehicle Communications," in *Proceedings of 2012 IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS)*. IEEE, 2012, pp. 328–333.

[63] K. Shilton, J. A. Burke, and L. Zhang, "Anticipating Policy and Social Implications of Named Data Networking," *Communication of the ACM*, vol. 59, no. 12, pp. 92–101, 2016.

[64] X. Jiang and J. Bi, "Interest Set Mechanism to Improve the Transport of Named Data Networking," in *Proceedings of the ACM SIGCOMM 2013 Conference on SIGCOMM - SIGCOMM '13*, no. 4. ACM, 2013, pp. 515–516.

[65] D. Perino, M. Varvello, L. Linguaglossa, R. Laufer, and R. Boislaigue, "Caesar: A Content Router for High-Speed Forwarding on Content Names," in *Proceedings of the tenth ACM/IEEE Symposium on Architectures for Networking and Communications Systems*. ACM, 2014, pp. 137–148.

[66] M. Dehghan, B. Jiang, A. Dabirmoghaddam, and D. Towsley, "On the Analysis of Caches with Pending Interest Tables," in *Proceedings of the 2nd International Conference on Information-Centric Networking*. ACM, 2015, pp. 69–78.

[67] Z. Li, K. Liu, Y. Zhao, and Y. Ma, "MaPIT : An Enhanced Pending Interest Table for NDN With Mapping Bloom Filter," *IEEE Communications Letters*, vol. 18, no. 11, pp. 1915–1918, 2014.

[68] W. You, B. Mathieu, P. Truong, J.-F. Peltier, and G. Simon, "DiPIT: A Distributed Bloom-Filter Based PIT Table for CCN Nodes," in *Proceedings of 2012 21st International Conference on Computer Communications and Networks (ICCCN)*. IEEE, 2012, pp. 1–7.

[69] C. Anastasiades, L. von Rotz, and T. Braun, "Adaptive Interest Lifetimes for Information-Centric Wireless Multi-hop Communication," in *Proceedings of 2015 8th IFIP Wireless and Mobile Networking Conference (WMNC)*. IEEE, 2015, pp. 40–47.

[70] G. Carofiglio, M. Gallo, and L. Muscariello, "ICP: Design and Evaluation of an Interest Control Protocol for Content-Centric Networking," in *Proceedings of 2012 IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS)*. IEEE, 2012, pp. 304–309.

[71] C. Yao, L. Fan, Z. Yan, and Y. Xiang, "Long-Term Interest for Realtime Applications in the Named Data Network," in *Proceedings of ACM AsiaFI12*. ACM, 2012, pp. 1–8.

[72] D. Grund and J. Reineke, "Estimating the Performance of Cache Replacement Policies," in *Proceedings of 6th ACM/IEEE International Conference on Formal Methods and Models for Co-Design, 2008 (MEMOCODE 2008)*. IEEE, 2008, pp. 101–112.

[73] Y. Smaragdakis, "General Adaptive Replacement Policies," in *Proceedings of the 4th International Symposium on Memory Management*. ACM, 2004, pp. 108–119.

[74] B. Azimdoost, C. Westphal, and H. R. Sadjadpour, "Fundamental Limits on Throughput Capacity in Information-Centric Networks," *IEEE Transactions on Communications*, vol. 64, no. 12, pp. 5037–5049, 2016.

[75] A. Zahemszky, B. Gajic, C. E. Rothenberg, C. Reason, D. Trossen, D. Lagutin, J. Tuononen, and K. Katsaros, "Experimentally-Driven Research in Publish/-Subscribe Information-Centric Inter-Networking," in *Proceedings of International Conference on Testbeds and Research Infrastructures*. Springer, 2010, pp. 469–485.

[76] H. Al-Zoubi, A. Milenkovic, and M. Milenkovic, "Performance Evaluation of Cache Replacement Policies for the SPEC CPU2000 Benchmark Suite," in *Proceedings of 42nd Annual Southeast Regional Conference*. ACM, 2004, pp. 267–272.

[77] K. Shimazu, N. Kawashima, and M. Ito, "Position paper: Design Concept of Ad-hoc Information Network System for disaster mitigation," in *Communications (APCC), 2013 19th Asia-Pacific Conference on*. IEEE, 2013, pp. 138–141.

[78] C. Yi, "Adaptive Forwarding in Named Data Networking," Ph.D. dissertation, University of Arizona - Graduate College, 2014.

[79] P. Rodriguez, S.-M. Tan, and C. Gkantsidis, "On the Feasibility of Commercial, Legal P2P Content Distribution," *ACM SIGCOMM Computer Communication Review*, vol. 36, no. 1, pp. 75–78, 2006.

[80] F. Almeida, M. T. Andrade, N. B. Melazzi, H. Walker, Richard nad Hussmann, and I. S. Venieris, *Enhancing the Internet with CONVERGENCE System*. Springer Science and Business Media, 2014, vol. Furth.

[81] P. Daras, D. Williams, C. Guerrero, I. Kegel, I. Laso, J. Bouwen, J. Meunier, N. Niebert, and T. Zahariadis, "Why do We Need a Content-Centric Future Internet," *Information Society and Media Journal*, pp. 1–23, 2009.

[82] M. Tortelli, D. Rossi, G. Boggia, and L. Grieco, "ICN Software Tools: Survey and Cross-Comparison," *Simulation Modelling Practice and Theory*, vol. 63, pp. 23–46, 2016.

[83] D. Saxena, V. Raychoudhury, and N. SriMahathi, "SmartHealth-NDNoT: Named Data Network of Things for Healthcare Services," in *MobileHealth 2015 Proceedings of the 2015 Workshop on Pervasive Wireless Healthcare*. ACM, 2015, pp. 1–6.

[84] G. Xylomenos, C. N. Ververidis, V. A. Siris, N. Fotiou, C. Tsilopoulos, X. Vasilakos, K. V. Katsaros, and G. C. Polyzos, "A Survey of Information-Centric Networking Research," *IEEE Communications Surveys and Tutorials*, vol. 16, no. 2, pp. 1024–1049, 2014.

[85] M. F. Bari, S. R. Chowdhury, R. Ahmed, R. Boutaba, and B. Mathieu, "A Survey of Naming and Routing in Information-Centric Networks," *IEEE Communications Magazine*, vol. 50, no. 12, pp. 44–53, 2012.

[86] B. Ahlgren, C. Dannewitz, C. Imbrenda, D. Kutscher, and B. Ohlman, "A Survey of Information-Centric Networking," *IEEE Communications Magazine*, vol. 50, no. 7, pp. 26–36, 2012.

[87] R. Tourani, T. Mick, S. Misra, and G. Panwar, "Security, Privacy, and Access Control in Information-Centric Networking: A Survey," *IEEE Communications Surveys and Tutorials*, pp. 1–35, 2016.

[88] D. Goergen, T. Cholez, J. Francois, and T. Engel, "Security Monitoring for Content-Centric Networking," in *Data Privacy Management and Autonomous Spontaneous Security*. Springer, 2013, pp. 274–286.

[89] G. Tyson, N. Sastry, I. Rimac, R. Cuevas, and A. Mauthe, "A Survey of Mobility in Information-Centric Networks : Challenges and Research Directions," in *Proceedings of the 1st ACM workshop on Emerging Name-Oriented Mobile Networking Design-Architecture, Algorithms, and Applications*. ACM, 2012, pp. 1–6.

[90] Y. Rekhter, T. Li, and S. Hares, "A Border Gateway Protocol 4 (BGP-4)," Network Working Group, Technical report, Tech. Rep., 2006.

[91] H. Wang, Z. Chen, F. Xie, and F. Han, "A Data Structure for Content Cache Management in Content-Centric Networking," in *Proceedings of 2012 Third International Conference on Networking and Distributed Computing*. IEEE, 2012, pp. 11–15.

[92] G. Carofiglio, V. Gehlen, and D. Perino, "Experimental Evaluation of Memory Management in Content-Centric Networking," in *Proceedings of 2011 IEEE International Conference on Communications (ICC)*. IEEE, 2011, pp. 1–6.

[93] C. Tsilopoulos and G. Xylomenos, "Supporting Diverse Traffic Types in Information Centric Networks," in *Proceedings of the ACM SIGCOMM Workshop on Information-Centric Networking*. ACM, 2011, pp. 13–18.

[94] A. Carzaniga, D. S. Rosenblum, and A. L. Wolf, "Content-Based Addressing and Routing: A General Model and its Application," University of Colorado, Technical Report CU-CS-902-00, Tech. Rep., 2000.

[95] A. Carzaniga, M. J. Rutherford, and A. L. Wolf, "A Routing Scheme for Content-based Networking," in *Proceedings of Twenty-third AnnualJoint Conference of the IEEE Computer and Communications Societies (INFOCOM)*. IEEE, 2004, pp. 918–928.

[96] B. Ahlgren, M. D Ambrosio, M. Marchisio, I. Marsh, C. Dannewitz, B. Ohlman, K. Pentikousis, O. Strandberg, R. Rembarz, and V. Vercellone, "Design Considerations for a Network of Information," in *Proceedings of the 2008 ACM CoNEXT Conference*. ACM, 2008, pp. 66–72.

[97] P. Jokela, A. Zahemszky, C. Esteve Rothenberg, S. Arianfar, and P. Nikander, "LIPSIN: Line Speed Publish/Subscribe Inter-Networking," *ACM SIGCOMM Computer Communication Review*, vol. 39, no. 4, pp. 195–206, 2009.

[98] V. Dimitrov and V. Koptchev, "PSIRP: Publish-Subscribe Internet Routing Paradigm: New Ideas for Future Internet," in *Proceedings of the 11th International Conference on Computer Systems and Technologies (CompSysTech10)*. ACM, 2010, pp. 167–171.

[99] G. Garcia, A. Beben, F. J. Ramon, A. Maeso, I. Psaras, G. Pavlou, N. Wang, J. Sliwinski, S. Spirou, S. Soursos *et al.*, "COMET: Content Mediator Architecture for Content-aware Networks," in *Proceedings of Future Network and

MobileSummit 2011 Conference Proceedings Paul Cunningham and Miriam Cunningham (Eds) IIMC International Information Management Corporation. IEEE, 2011, pp. 1–8.

[100] K. Katsaros, G. Xylomenos, and G. C. Polyzos, "MultiCache: An Overlay Architecture for Information-Centric Networking," *Computer Networks*, vol. 55, no. 4, pp. 936–947, 2011.

[101] A. Detti, N. Blefari Melazzi, S. Salsano, and M. Pomposini, "CONET: A Content Centric Inter-Networking Architecture," in *Proceedings of the ACM SIGCOMM Workshop on Information-Centric Networking*. ACM, 2011, pp. 50–55.

[102] D. Raychaudhuri, K. Nagaraja, and A. Venkataramani, "MobilityFirst: A Robust and Trustworthy Mobility-Centric Architecture for the Future Internet," *ACM SIGMOBILE Mobile Computing and Communications Review*, vol. 16, no. 3, pp. 2–13, 2012.

[103] M. Amadeo, A. Molinaro, and G. Ruggeri, "E-CHANET: Routing, Forwarding and Transport in Information-Centric Multihop Wireless Networks," *Computer communications*, vol. 36, no. 7, pp. 792–803, 2013.

[104] Y. Wang, K. He, H. Dai, W. Meng, J. Jiang, B. Liu, and Y. Chen, "Scalable Name Lookup in NDN Using Effective Name Component Encoding," in *Proceedings of 2012 IEEE 32nd International Conference on Distributed Computing Systems (ICDCS)*. IEEE, 2012, pp. 688–697.

[105] C. Yi, J. Abraham, A. Afanasyev, L. Wang, B. Zhang, and L. Zhang, "On The Role of Routing in Named Data Networking," in *Proceedings of the 1st International Conference on Information-Centric Networking*. ACM, 2014, pp. 27–36.

[106] A. Hoque, S. O. Amin, A. Alyyan, B. Zhang, L. Zhang, and L. Wang, "NLSR : Named-data Link State Routing Protocol," in *Proceedings of the 3rd ACM SIGCOMM Workshop on Information-Centric Networking*. ACM, 2013, pp. 15–20.

[107] H. Dai, J. Lu, Y. Wang, and B. Liu, "A Two-layer Intra-domain Routing Scheme for Named Data Networking," in *Proceedings of 2012 IEEE Global Communications Conference (GLOBECOM)*. IEEE, 2012, pp. 2815–2820.

[108] L. Wang, A. Hoque, C. Yi, A. Alyyan, and B. Zhang, "OSPFN: An OSPF based Routing Protocol for Named Data Networking," University of Memphis and University of Arizona, Technical Report NDN-0003, Tech. Rep., 2012.

[109] H. Choi, J. Yoo, T. Chung, N. Choi, T. Kwon, and Y. Choi, "CoRC: Coordinated Routing and Caching for Named Data Networking," in *Proceedings of the tenth ACM/IEEE Symposium on Architectures for Networking and Communications Systems*. ACM, 2014, pp. 161–172.

[110] X. Hu, C. Papadopoulos, J. Gong, and D. Massey, "Not So Cooperative Caching in Named Data Networking," in *Proceedings of 2013 IEEE Global Communications Conference (GLOBECOM)*. IEEE, 2013, pp. 2263–2268.

[111] M. Rezazad and Y. Tay, "A Cache Miss Equation for Partitioning an NDN Content Store," in *Proceedings of the 9th Asian Internet Engineering Conference*. ACM, 2013, pp. 1–8.

[112] H. Dai, Y. Wang, H. Wu, J. Lu, and B. Liu, "Towards Line-Speed and Accurate On-line Popularity Monitoring on NDN Routers," in *Proceedings of 2014 IEEE 22nd International Symposium of Quality of Service (IWQoS)*. IEEE, 2014, pp. 178–187.

[113] J. Ran, N. Lv, D. Zhang, Y. Ma, and Z. Xie, "On Performance of Cache Policies in Named Data Networking," in *Proceedings of International Conference on Advanced Computer Science and Electronics Information*, 2013, pp. 668–671.

[114] W. Dron, A. Leung, M. Uddin, S. Wang, T. Abdelzaher, R. Govindan, and J. Hancock, "Information-maximizing Caching in Ad Hoc Networks with Named Data Networking," in *Proceedings of 2013 IEEE 2nd Network Science Workshop (NSW)*. IEEE, 2013, pp. 90–93.

[115] S. H. Ahmed, S. H. Bouk, M. A. Yaqub, D. Kim, H. Song, and J. Lloret, "CODIE: COntrolled Data and Interest Evaluation in Vehicular Named Data Networks," *IEEE Transactions on Vehicular Technology*, vol. 65, no. 6, pp. 3954–3963, 2016.

[116] S. H. Bouk, M. Yaqub, S. H. Ahmed, and D. Kim, "Evaluating Interest/Data Propagation in Vehicular Named Data Networks," in *Proceedings of the 2015 Conference on research in adaptive and convergent systems*. ACM, 2015, pp. 256–259.

[117] Y. Rao, D. Gao, H. Zhang, and C. H. Foh, "Mobility Support for the User in NDN-Based Cloud Storage Service," in *Proceedings of 2015 IEEE Globecom Workshops*. IEEE, 2015, pp. 1–6.

[118] X. Jiang, J. Bi, and Y. Wang, "What Benefits Does NDN Have in Supporting Mobility," in *Proceedings of 2014 IEEE Symposium on Computers and Communications (ISCC)*. IEEE, 2014, pp. 1–6.

[119] L. Wang, O. Waltari, and J. Kangasharju, "MobiCCN: Mobility Support with Greedy Routing in Content-Centric Networks," in *Proceedings of 2013 IEEE Global Communications Conference (GLOBECOM)*. IEEE, 2013, pp. 2069–2075.

[120] A. A. Barakabitze, T. Xiaoheng, and G. Tan, "A Survey on Naming, Name Resolution and Data Routing in Information Centric Networking (ICN)," *International Journal of Advanced Research in Computer and Communication Engineering*, vol. 3, no. 10, pp. 8322–8330, 2014.

[121] A. Afanasyev, "Addressing Operational Challenges in Named Data Networking Through NDNs Distributed Database," Ph.D. dissertation, University of California, 2013.

[122] R. L. Rivest, A. Shamir, and L. Adleman, "A Method for Obtaining Digital Signatures and Public-Key Cryptosystems," *Communications of the ACM*, vol. 21, no. 2, pp. 120–126, 1978.

[123] M. Gao, X. Zhu, and Y. Su, "Protecting Router Cache Privacy in Named Data Networking," in *Proceedings of 2015 IEEE/CIC International Conference on Communications in China (ICCC)*. IEEE, 2015, pp. 1–5.

[124] T. Nguyen, R. Cogranne, and G. Doyen, "An Optimal Statistical Test for Robust Detection against Interest Flooding Attacks in CCN," in *Proceedings of 2015 IFIP/IEEE International Symposium on Integrated Network Management (IM)*. IEEE, 2015, pp. 252–260.

[125] C. Ghali, G. Tsudik, and E. Uzun, "Network-Layer Trust in Named-Data Networking," *ACM SIGCOMM Computer Communication Review*, vol. 44, no. 5, pp. 12–19, 2014.

[126] D. Kim, S. Nam, J. Bi, and I. Yeom, "Efficient Content Verification in Named Data Networking," in *Proceedings of the 2nd International Conference on Information-Centric Networking*. ACM, 2015, pp. 109–116.

[127] A. Compagno, M. Conti, C. Ghali, and G. Tsudik, "To NACK or not to NACK? Negative Acknowledgments in Information-Centric Networking," in *Proceedings of 2015 24th International Conference on Computer Communication and Networks (ICCCN)*. IEEE, 2015, pp. 1–10.

[128] T. Asami, B. Namsraijav, Y. Kawahara, K. Sugiyama, A. Tagami, T. Yagyu, K. Nakamura, and T. Hasegawa, "Moderator-Controlled Information Sharing by Identity-Based Aggregate Signatures for Information Centric Networking," in *Proceedings of the 2nd International Conference on Information-Centric Networking*. ACM, 2015, pp. 157–166.

[129] T. Song, H. Yuan, P. Crowley, and B. Zhang, "Scalable Name-Based Packet Forwarding: From Millions to Billions," in *Proceedings of the 2nd International Conference on Information-Centric Networking*. ACM, 2015, pp. 19–28.

[130] Y. Li, D. Zhang, X. Yu, W. Liang, J. Long, and H. Qiao, "Accelerate NDN Name Lookup using FPGA: Challenges and A Scalable Approach," in *Proceedings of 24th International Conference on Field Programmable Logic and Applications (FPL)*. IEEE, 2014, pp. 1–4.

[131] D. O Coileain and D. O Mahony, "SAVANT: Aggregated Feedback and Accountability Framework for Named Data Networking," in *Proceedings of the 1st international conference on Information-Centric Networking*. ACM, 2014, pp. 187–188.

[132] A. Compagno, M. Conti, P. Gasti, L. V. Mancini, and G. Tsudik, "Violating Consumer Anonymity: Geo-locating Nodes in Named Data Networking," in *Proceedings of International Conference on Applied Cryptography and Network Security*. Springer, 2015, pp. 243–262.

[133] X. Zeng and Z. H. Gao, "Rank-Based Routing Strategy for Named Data Network," in *Proceedings of Applied Mechanics and Materials*. Trans Tech Publications, Switzerland, 2014, pp. 3320–3323.

[134] M. Tortelli, L. A. Grieco, and G. Boggia, "Performance Assessment of Routing Strategies in Named Data Networking," in *Proceedings of GTTI 2013 Session on Telecommunication Networks*. IEEE, 2013, pp. 1–6.

[135] L. Wang, A. Afanasyev, R. Kuntz, R. Vuyyuru, R. Wakikawa, and L. Zhang, "Rapid Traffic Information Dissemination Using Named Data," in *Proceedings of the 1st ACM workshop on Emerging Name-Oriented Mobile Networking Design-Architecture, Algorithms, and Applications*. ACM, 2012, pp. 7–12.

[136] M. Varvello, I. Rimac, U. Lee, L. Greenwald, and V. Hilt, "On the Design of Content-Centric MANETs," in *Proceedings of 2011 Eighth International Conference on Wireless On-Demand Network Systems and Services (WONS)*. IEEE, 2011, pp. 1–8.

[137] Y.-T. Yu, R. B. Dilmaghani, S. Calo, M. Sanadidi, and M. Gerla, "Interest Propagation in Named Data MANETs," in *Proceedings of 2013 International Conference on Computing, Networking and Communications (ICNC)*. IEEE, 2013, pp. 1118–1122.

[138] M. Meisel, V. Pappas, and L. Zhang, "Ad Hoc Networking via Named Data," in *Proceedings of the Fifth ACM International Workshop on Mobility in the Evolving Internet Architecture*. ACM, 2010, pp. 3–8.

[139] P. Gusev and J. Burke, "NDN-RTC: Real-Time Videoconferencing over Named Data Networking," in *Proceedings of the 2nd International Conference on Information-Centric Networking*. ACM, 2015, pp. 117–126.

[140] G. Piro, V. Ciancaglini, R. Loti, L. A. Grieco, and L. Liquori, "Providing Crowd-Sourced and Real-Time Media Services through an NDN-Based Platform," in *Modeling and Processing for Next-Generation Big-Data Technologies*. Springer, 2015, pp. 405–441.

[141] D. Posch, C. Kreuzberger, B. Rainer, and H. Hellwagner, "Client Starvation: A Shortcoming of Client-driven Adaptive Streaming in Named Data Networking," in *Proceedings of the 1st International Conference on Information-Centric Networking*. ACM, 2014, pp. 183–184.

[142] J. Lindblom, M. Huang, J. Burke, and L. Zhang, "FileSync/NDN: Peer-to-Peer File Sync over Named Data Networking," NDN Technical Report NDN-0012, Tech. Rep., 2013, accessed: 13-Apr-2015. [Online]. Available: https://named-data.net/wp-content/uploads/TRFileSync.pdf

[143] Z. Zhu and A. Afanasyev, "Let's Chronosync: Decentralized Dataset State Synchronization in Named Data Networking," in *Proceedings of 2013 21st IEEE International Conference on Network Protocols (ICNP)*. IEEE, 2013, pp. 1–10.

[144] M. Amadeo, C. Campolo, and A. Molinaro, "Multi-Source Data Retrieval in IoT via Named Data Networking," in *Proceedings of the 1st International Conference on Information-Centric Networking*. ACM, 2014, pp. 67–76.

[145] S. Wang, J. Wu, and J. Bi, "Application Design over Named Data Networking with its Features in Mind," in *Proceedings of the 11th International Conference on Networks (ICN)*, 2012, pp. 121–124.

[146] D. Massey, C. Papadopoulos, L. Wang, B. Zhang, and L. Zhang, "Teaching Network Architecture through Case Studies," in *Proceedings of ACM SIGCOMM Education Workshop*. ACM, 2011, pp. 1–5.

[147] M. Chen, "NDNC-BAN: Supporting Rich Media Healthcare Services via Named Data Networking in Cloud-assisted Wireless Body Area Networks," *Information Sciences*, vol. 284, pp. 142–156, 2014.

[148] B. Etefia and L. Zhang, "Named Data Networking for Military Communication Systems," in *Proceedings of 2012 IEEE Aerospace Conference*. IEEE, 2012, pp. 1–7.

[149] Z. Qu and J. Burke, "Egal Car: A Peer-to-Peer Car Racing Game Synchronized Over Named Data Networking," NDN, Technical Report NDN-0010, Tech. Rep., 2012.

[150] S. Y. Oh, D. Lau, and M. Gerla, "Content Centric Networking in Tactical and Emergency MANETs," in *Wireless Days (WD) - 2010 IFIP*. IEEE, 2010, pp. 1–5.

[151] O. Briante, M. Amadeo, C. Campolo, A. Molinaro, S. Y. Paratore, and G. Ruggeri, "eDomus: User-home Interactions through Facebook and Named Data Networking," in *Proceedings of 2014 Eleventh Annual IEEE International Conference on Sensing, Communication, and Networking (SECON)*. IEEE, 2014, pp. 155–157.

[152] C. Bernardini, T. Silverston, and O. Festor, "Using Social Network Information into ICN," HAL, Technical report, Tech. Rep., 2013.

[153] G. Grassi, D. Pesavento, G. Pau, R. Vuyyuru, R. Wakikawa, and L. Zhang, "VANET via Named Data Networking," in *Proceedings of 2014 IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS)*. IEEE, 2014, pp. 410–415.

[154] I. Moiseenko, L. Wang, and L. Zhang, "Consumer/Producer Communication with Application Level Framing in Named Data Networking," in *Proceedings of the 2nd International Conference on Information-Centric Networking*. ACM, 2015, pp. 99–108.

[155] I. Moiseenko and L. Zhang, "Consumer-Producer API for Named Data Networking," in *Proceedings of the 1st International Conference on Information-Centric Networking*. ACM, 2014, pp. 177–178.

[156] D. G. Barros and M. P. Fernandez, "NDNGame: A NDN-based Architecture for Online Games," in *Proceedings of the 2nd ACM International Conference on Information-centric Networking*. ACM, 2015, pp. 65–72.

[157] Z. Wang, Z. Qu, and J. Burke, "Matryoshka: Design of NDN Multiplayer Online Game," in *Proceedings of the 1st International Conference on Information-Centric Networking*. ACM, 2014, pp. 209–210.

[158] A. Attam and I. Moiseenko, "NDNBlue : NDN over Bluetooth," NDN, Technical report, Tech. Rep., 2013, accessed: 13-Jan-2015. [Online]. Available: http://named-data.net/wp-content/uploads/2013/07/TR-NDN-0015-NDNBlue.pdf

[159] S. Shannigrahi, C. Papadopoulos, E. Yeh, H. Newman, A. J. Barczyk, R. Liu, A. Sim, A. Mughal, I. Monga, J.-R. Vlimant *et al.*, "Named Data Networking in Climate Research and HEP Applications," in *21st International Conference on Computing in High Energy and Nuclear Physics (CHEP2015).* IOP Publishing, 2015, pp. 1–8, accessed: 7-Feb.-2016. [Online]. Available: http://iopscience.iop.org/article/10.1088/1742-6596/664/5/052033/pdf

[160] M. Gallo, L. Gu, D. Perino, and M. Varvello, "NaNET : Socket API and Protocol Stack for Process-to-Content Network Communication," in *Proceedings of the 1st International Conference on Information-Centric Networking*. ACM, 2014, pp. 185–186.

[161] S. H. Bouk, S. H. Ahmed, D. Kim, and H. Song, "Named-Data-Networking-Based ITS for Smart Cities," *IEEE Communications Magazine*, vol. 55, no. 1, pp. 105–111, 2017.

[162] G. Piro, I. Cianci, L. A. Grieco, G. Boggia, and P. Camarda, "Information Centric Services in Smart Cities," *Journal of Systems and Software*, vol. 88, pp. 169–188, 2014.

[163] A. A. Alsaffar and E.-N. Huh, "A Framework of N-Screen Services based on PVR and Named Data Networking in Cloud Computing," in *Proceedings of the 7th International Conference on Ubiquitous Information Management and Communication*. ACM, 2013, pp. 1–7.

[164] H. D. Bandara and A. P. Jayasumana, "Distributed, Multi-user, Multi-application, and Multi-sensor Data Fusion over Named Data Networks," *Computer Networks*, vol. 57, no. 16, pp. 3235–3248, 2013.

[165] T. Ogawara, Y. Kawahara, and T. Asami, "Information Dissemination Performance of a Disaster-tolerant NDN-based Distributed Application in Disrupted Cellular Networks," in *Proceedings of 13th IEEE International Conference on Peer-to-Peer Computing*. IEEE, 2013, pp. 1–5.

[166] K. Wang, H. Zhou, J. Chen, and Y. Qin, “RDAI: Router-based Data Aggregates Identification Mechanism for Named Data Networking,” in *Proceedings of 2013 Seventh International Conference on Innovative Mobile and Internet Services in Ubiquitous Computing (IMIS)*. IEEE, 2013, pp. 116–121.

[167] Y. Song, M. Liu, and Y. Wang, “Power-aware Traffic Engineering with Named Data Networking,” in *Proceedings of 2011 Seventh International Conference on Mobile Ad-hoc and Sensor Networks (MSN)*. IEEE, 2011, pp. 289–296.

[168] M. Zhang, C. Yi, B. Liu, and B. Zhang, “GreenTE: Power-aware Traffic Engineering,” in *Proceedings of 2010 18th IEEE International Conference on Network Protocols (ICNP)*. IEEE, 2010, pp. 21–30.

[169] J. Wang, C. Peng, C. Li, E. Osterweil, R. Wakikawa, P.-c. Cheng, and L. Zhang, “Implementing Instant Messaging Using Named Data,” in *Proceedings of the Sixth Asian Internet Engineering Conference*. ACM, 2010, pp. 40–47.

[170] A. J. Abu, B. Bensaou, and J. M. Wang, “Interest Packets Retransmission in Lossy CCN Networks and its Impact on Network Performance,” in *Proceedings of the 1st International Conference on Information-Centric Networking*. ACM, 2014, pp. 167–176.

[171] R. Alubady, S. Hassan, and A. Habbal, “A Taxonomy of Pending Interest Table Implementation Approaches in Named Data Networking,” *Journal of Theoretical and Applied Information Technology*, vol. 91, no. 2, pp. 411–423, 2016.

[172] C. Protocol, “CCNx Protocol,” Technical documentation for ccnx 0.8.2, accessed: 5-Sep-2013. [Online]. Available: http://www.ccnx.org/releases/ccnx-0.8.2/doc/technical/CCNxProtocol.html

[173] H. Yuan, T. Song, and P. Crowley, “Scalable NDN Forwarding: Concepts, Issues and Principles,” in *Proceedings of 2012 21st International Conference on Computer Communications and Networks (ICCCN)*. IEEE, 2012, pp. 1–9.

[174] G. Ma, Z. Chen, and K. Zhao, “A Cache Management Strategy for Content Store in Content Centric Network,” in *Proceedings of 2013 Fourth International Conference on Networking and Distributed Computing*. IEEE, 2013, pp. 94–99.

[175] G. Damien, “STUDY OF DIFFERENT CACHE LINE REPLACEMENT ALGORITHMS IN EMBEDDED SYSTEMS,” Master’s thesis, ARM France SAS Les Cardoulines B2 - Route des Dolines Sophia Antipolis, 2007, accessed: 14-Sep.-2013. [Online]. Available: http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.217.3594&rep=rep1&type=pdf

[176] K. Arora and D. R. Ch, “Web Cache Page Replacement by Using LRU and LFU Algorithms with Hit Ratio: A Case Unification,” *(IJCSIT) International Journal of Computer Science and Information Technologies*, vol. 5, no. 3, pp. 3232–3235, 2014.

[177] P. R. Jelenkovic and X. Kang, "Characterizing the Miss Sequence of the LRU Cache," *ACM SIGMETRICS Performance Evaluation Review*, vol. 36, no. 2, pp. 119–121, 2008.

[178] I. Ud Din, S. Hassan, and A. Habbal, "Comparison of Caching Strategies on Different Topologies in Information-Centric Networking," in *Proceedings of the 4th International Conference on Internet Applications, Protocols and Services (NETAPPS2015) Comparison*. UUM, 2015, pp. 38–41.

[179] C. Ghali, G. Tsudik, E. Uzun, and C. A. Wood, "Living in a PIT-less World: A Case Against Stateful Forwarding in Content-Centric Networking," *arXiv preprint arXiv:1512.07755*, pp. 1–10, 2015.

[180] M. Allman and V. Paxson, "Computing TCP Retransmission Timer," Network Working Group-RFC 2988, Network Working Group-RFC 2988, Tech. Rep., 2000.

[181] R. Jain, *The Art of Computer Systems Performance Analysis: Techniques for Experimental Design, Measurement, Simulation, and Modeling*. Wiley Computer Publishing, John Wiley and Sons, 1990.

[182] V. V. Kalashnikov, *Mathematical Methods in Queuing Theory*. Springer Science and Business Media, 1993, vol. 271.

[183] R. Adams, "Active Queue Management: A Survey," *IEEE Communications Surveys and Tutorials*, vol. 15, no. 3, pp. 1425–1476, 2013.

[184] G. Thiruchelvi and J. Raja, "A Survey On Active Queue Management Mechanisms," *International Journal of Computer Science and Network Security*, vol. 8, no. 12, pp. 130–145, 2008.

[185] Y. Jiang, M. Hamdi, and J. Liu, "Self Adjustable CHOKe: An Active Queue Management Algorithm For Congestion Control and Fair Bandwidth Allocation," in *Proceedings of the Eighth IEEE Symposium on Computers and Communications (ISCC 2003)*. IEEE, 2003, pp. 1018–1025.

[186] J. Koo, B. Song, K. Chung, H. Lee, and H. Kahng, "MRED: A New Approach to Random Early Detection," in *Proceedings of 15th International Conference on Information Networking*. IEEE, 2001, pp. 347–352.

[187] S. Floyd, "TCP and Explicit Congestion Notification," *ACM SIGCOMM Computer Communication Review*, vol. 24, no. 5, pp. 8–23, 1994.

[188] C. Long, B. Zhao, X. Guan, and J. Yang, "The Yellow Active Queue Management Algorithm," *Computer Networks*, vol. 47, no. 4, pp. 525–550, 2005.

[189] S. Floyd and V. Jacobson, "Random Early Detection Gateways for Congestion Avoidance," *IEEE/ACM Transactions on Networking (ToN)*, vol. 1, no. 4, pp. 397–413, 1993.

[190] T. J. Ott, T. Lakshman, and L. H. Wong, "SRED: Stablised RED," in *INFOCOM'99. Eighteenth Annual Joint Conference of the IEEE Computer and Communications Societies. Proceedings. IEEE*. IEEE, 1999, pp. 1346–1355.

[191] S. S. Kunniyur and R. Srikant, "An Adaptive Virtual Queue (AVQ) Algorithm for Active Queue Management," *IEEE/ACM Transactions on Networking (ToN)*, vol. 12, no. 2, pp. 286–299, 2004.

[192] C.-N. Long, B. Zhao, and X.-P. Guan, "SAVQ : stabilized Adaptive Virtual Queue Management Algorithm," *IEEE Communications Letters*, vol. 9, no. 1, pp. 78–80, 2005.

[193] S. Athuraliya, S. H. Low, V. H. Li, and Q. Yin, "REM: Active Queue Management," *IEEE Network*, vol. 15, no. 3, pp. 48–53, 2001.

[194] X. Deng, S. Yi, G. Kesidis, and C. R. Das, "Stabilised Virtual Buffer (SVB)-An Active Queue Management Scheme for Internet Quality of Service," in *Global Telecommunications Conference, 2002. GLOBECOM'02. IEEE*. IEEE, 2002, pp. 1628–1632.

[195] E.-C. Park, H. Lim, K.-J. Park, and C.-H. Choi, "Analysis and Design of the Virtual rate Control Algorithm for Stabilizing Queues in TCP Networks," *Computer Networks*, vol. 44, no. 1, pp. 17–41, 2004.

[196] S. S. Kunniyur and R. Srikant, "Stable, Scalable, Fair Congestion Control and AQM Schemes That Achieve High Utilization in the Internet," *IEEE Transactions on Automatic Control*, vol. 48, no. 11, pp. 2024–2028, 2003.

[197] X. Deng, S. Yi, G. Kesidis, and C. R. Das, "Stabilized virtual buffer (SVB) - An Active Queue Management Scheme for Internet Quality-of-Service," in *Proceedings of Global Telecommunications Conference, 2002. GLOBECOM'02*. IEEE, 2002, pp. 1628–1632.

[198] Y. S. A. Miaji, "Just Queuing: Policy-Based Scheduling Mechanism for Packet Switching Networks," Ph.D. dissertation, Universiti Utara Malaysia, 2011.

[199] C. Couch, *Urban renewal: theory and practice*. Macmillan, 1990.

[200] G. Chandrasekaran, "Performance Evaluation of Scheduling Mechanisms for Broadband Networks," Ph.D. dissertation, University of Kansas, 2001, accessed: 16-Apr-2014. [Online]. Available: http://www.ittc-ku.net/research/thesis/documents/gayathri_chandrasekaran_thesis.pdf

[201] M. Pinedo, *Scheduling: Theory, Algorithms, and Systems*. Springer, 2015.

[202] H. Omar Mohd, "An Innovative Signal Detection Algorithm in Facilitating the Cognitive Radio Functionality for Wireless Regional Area Network Using Singular Value Decomposition," Ph.D. dissertation, Universiti Utara Malaysia, 2011.

[203] S. Hassan, W. Elbreiki, M. Firdhous, and A. M. M. Habbal, "End-to-End Networks vs Named Data Network: A Critical Evaluation," *Jurnal Teknologi*, vol. 72, no. 5, pp. 71–76, 2015.

[204] X. Chang, "Network Simulations with OPNET," in *Proceedings of the 31st Conference on Winter Simulation: Simulation a Bridge to the Future*. ACM, 1999, pp. 307–314.

[205] A. Varga, *OMNeT++*. Springer, 2010, ch. Modeling and Tools for Network Simulation, pp. 35–59.

[206] A. Varga and R. Hornig, "An Overview of the OMNeT++ Simulation Environment," in *Proceedings of the 1st International Conference on Simulation Tools and Techniques for Communications, Networks and Systems and Workshops*. ICST (Institute for Computer Sciences, Social-Informatics and Telecommunications Engineering), 2008, pp. 1–10.

[207] NS-3, "NS-3 Networking Simulator NS-3 Model Library," Tech. Rep., 2016. [Online]. Available: https://www.nsnam.org/docs/models/ns-3-model-library.pdf

[208] G. F. Riley and T. R. Henderson, *The NS-3 Network Simulator*. Springer, 2010, ch. Modeling and Tools for Network Simulation, pp. 15–34.

[209] G. F. Lucio, M. Paredes-Farrera, E. Jammeh, M. Fleury, and M. J. Reed, "OPNET Modeler and NS-2: Comparing the Accuracy of Network Simulators for Packet-Level Analysis Using a Network Testbed," *WSEAS Transactions on Computers*, vol. 2, no. 3, pp. 700–707, 2003.

[210] OMNeT++, "OMNeT++," June 2015, accessed:14-Jan-2015. [Online]. Available: https://omnetpp.org/

[211] J. Pan and R. Jain, "A Survey of Network Simulation Tools: Current Status and Future Developments," Washington University in St. Louis, Technical Report, Tech. Rep., 2008.

[212] C. Rajan, K. Geetha, C. Priya, S. Geetha, and A. Manikansan, "A Simple Analysis on Novel Based Open Source Network Simulation Tools for Mobile Ad Hoc Networks," *International Journal of Advanced Research in Computer Science and Software Engineering*, vol. 3, no. 9, pp. 856–862, 2013.

[213] W. Abduladheem, R. Alubady, and K. Nor, Shahrudin Awang, "Simulation-based Performance of Transport Protocols Using MPEG-4 Traffics over 4G Network," in *Proceedings of the 4th International Conference on Internet Applications, Protocols and Services (NETAPPS2015)*. UUM, 2015, pp. 116–122.

[214] K. Pentikousis, B. Ohlman, D. Corujo, G. Boggia, G. Tyson, E. Davies, A. Molinaro, and S. Eum, "Information-Centric Networking: Baseline Scenarios," Internet Research Task Force (IRTF) - draft-irtf-rrg- design-goals, Tech. Rep., 2015.

[215] K. Pentikousis, B. Ohlman, D. Corujo, G. Boggia, G. Tyson, E. Davies, P. Mahadevan, S. Spirou, A. Molinaro, D. Gellert *et al.*, "ICN Baseline Scenarios and Evaluation Methodology," *ICNRG-Draft-Pentikousis-ICN-Scenarios-04*, pp. 1–64, 2013.

[216] CCNPL-Sim, "CCN Packet Level Simulator (CCNPL-Sim)," accessed: 16-Feb-2011. [Online]. Available: http://systemx.enst.fr/ccnpl-sim

[217] R. Chiocchetti, D. Rossi, and G. Rossini, "ccnSim: An Highly Scalable CCN Simulator," in *IEEE International Conference on Communications*. IEEE, 2013, pp. 2309–2314.

[218] S. Mastorakis, A. Afanasyev, I. Moiseenko, and L. Zhang, "ndnSIM 2.0: A New Version of the NDN Simulator for NS-3," University of California, Los Angeles, NDN, Technical ReportNDN-0028, Tech. Rep., 2015.

[219] Parc, "The CCNx Project," accessed: 24-Jun-2013. [Online]. Available: http://blogs.parc.com/ccnx/

[220] I. Abdullahi, "Proxcache: A New Cache Deployment Strategy in Information-Centric Network for Mitigating Path and Content Redundancy," Ph.D. dissertation, University Utara Malaysia, 2016.

[221] W. Wang, Y. Sun, Y. Guo, D. Kaafar, J. Jin, J. Li, and Z. Li, "CRCache: Exploiting the Correlation between Content Popularity and Network Topology Information for ICN Caching," in *Communications (ICC), 2014 IEEE International Conference on*. IEEE, 2014, pp. 3191–3196.

[222] X. Vasilakos, V. A. Siris, G. C. Polyzos, and M. Pomonis, "Proactive Selective Neighbor Caching for Enhancing Mobility Support in Information-Centric Networks," in *Proceedings of the second edition of the ICN workshop on Information-centric networking*. ACM, 2012, pp. 61–66.

[223] ndnSIM, "Overall ndnSIM Documentation: Obtaining Metrics," accessed: 26-Sep-2013. [Online]. Available: http://ndnsim.net/1.0/metric.html

[224] M. Mangili, F. Martignon, and A. Capone, "A Comparative Study of Content-Centric and Content-Distribution Networks: Performance and Bounds," in *Proceedings of 2013 IEEE Global Communications Conference (GLOBECOM)*. IEEE, 2013, pp. 1403–1409.

[225] S. A. Nor, S. Hassan, O. Ghazali, M. M. Kadhum, and M. H. Omar, "Performance Enhancement of DCCP TCP-like over Long Delay Link Networks," *International Journal of Modeling and Optimization*, vol. 2, no. 3, p. 315, 2012.

[226] Y. Kim, Y. Kim, J. Bi, and I. Yeom, "Differentiated Forwarding and Caching in Named-Data Networking," *Journal of Network and Computer Applications*, vol. 60, pp. 155–169, 2016.

[227] Y. Wang, N. Rozhnova, A. Narayanan, D. Oran, and I. Rhee, "An Improved Hop-by-Hop Interest Shaper for Congestion Control in Named Data Networking," *ACM SIGCOMM Computer Communication Review*, vol. 43, no. 4, pp. 55–60, 2013.

[228] J. Li, H. Wu, B. Liu, J. Lu, Y. Wang, X. Wang, Y. Zhang, and L. Dong, "Popularity-driven Coordinated Caching in Named Data Networking," in *Proceedings of the eighth ACM/IEEE symposium on Architectures for networking and communications systems*. ACM, 2012, pp. 15–26.

[229] A. Laszka and A. Gueye, "Network topology vulnerability/cost trade-off: Model, application, and computational complexity," *Internet Mathematics*, vol. 11, no. 6, pp. 588–626, 2015.

[230] M. Mangili, F. Martignon, and A. Capone, "Performance Analysis of Content-Centric and Content-Delivery Networks with Evolving Object Popularity," *Computer Networks*, vol. 94, pp. 80–98, 2016.

[231] G. Rossini and D. Rossi, "Evaluating CCN Multi-path Interest Forwarding Strategies," *Computer Communications*, vol. 36, no. 7, pp. 771–778, 2013.

[232] R. Mahajan, N. Spring, D. Wetherall, and T. Anderson, "Inferring Link Weights using End-to-End Measurements," in *Proceedings of the 2nd ACM SIGCOMM Workshop on Internet measurment*. ACM, 2002, pp. 231–236.

[233] A. J. Abu and B. Bensaou, "Modelling the Pending Interest Table occupancy in CCN with interest timeout and retransmission," in *2015 IEEE 40th Conference on Local Computer Networks (LCN)*. IEEE, 2015, pp. 37–45.

[234] K. Wang, H. Zhou, Y. Qin, J. Chen, and H. Zhang, "Decoupling malicious Interests from Pending Interest Table to Mitigate Interest Flooding Attacks," in *Proceedings of 2013 IEEE Globecom Workshops*. IEEE, 2013, pp. 963–968.

[235] D. Rossi and G. Rossini, "Caching Performance of Content Centric Networks under Multi-path Routing (and more)," *Relatorio tecnico, Telecom ParisTech*, pp. 1–9, 2011.

[236] Q. Li, Z. Zhao, M. Xu, Y. Jiang, and Y. Yang, "A Smart Routing Scheme for Named Data Networks," *Computer Communications*, pp. 1–35, 2016.

[237] J. Shi and B. Zhang, "NDNLP : A Link Protocol for NDN," The University of Arizona, Tucson, AZ, NDN Technical Report NDN-0006, Tech. Rep., 2012.

[238] M. Amadeo and A. Molinaro, "CHANET: A Content-Centric Architecture for IEEE 802.11 MANETs," in *Proceedings of 2011 International Conference on the Network of the Future (NOF 2011)*. IEEE, 2011, pp. 122–127.

[239] M. Rezazad and Y. Tay, "CCndnS: A Strategy for Spreading Content and Decoupling NDN Caches," in *Proceedings of IFIP Networking Conference (IFIP Networking)*. IEEE, 2015, pp. 1–9.

[240] S. Tarnoi, K. Suksomboon, W. Kumwilaisak, and Y. Ji, "Performance of Probabilistic Caching and Cache Replacement Policies for Content-Centric Networks," in *39th Annual IEEE Conference on Local Computer Networks*. IEEE, 2014, pp. 99–106.

[241] E. Yeh, R. Liu, T. Ho, M. Burd, Y. Cui, and D. Leong, "Forwarding, Caching and Congestion Control in Named Data Networks," in *Proceedings of the 1st International Conference on Information-Centric Networking*, 2013, pp. 1–19.

[242] H. Wu, J. Li, T. Pan, and B. Liu, "A Novel Caching Scheme for the Backbone of Named Data Networking," in *Proceedings of 2013 IEEE International Conference on Communications (ICC)*. IEEE, 2013, pp. 3634–3638.

[243] H. Wu, J. Li, Y. Wang, and B. Liu, "EMC: The Effective Multi-path Caching Scheme for Named Data Networking," in *Proceedings of 2013 22nd International Conference on Computer Communication and Networks (ICCCN)*. IEEE, 2013, pp. 1–7.

[244] S. Shailendra, S. Sengottuvelan, H. K. Rath, B. Panigrahi, and A. Simha, "Performance Evaluation of Caching Policies in NDN-an ICN Architecture," *arXiv preprint arXiv:1612.00352*, pp. 1–5, 2016.

[245] A. Suki, S. Hassan, and I. Abdullahi, "Cache Replacement Positions in Information-Centric Network," in *Proceedings of the 4th International Conference on Internet Applications, Protocols and Services (NETAPPS2015)*. UUM, 2015, pp. 54–58.

[246] S. Gao and H. Zhang, "Scalable Mobility Management for Content Sources in Named Data Networking," in *Proceedings of 2016 13th IEEE Annual Consumer Communications and Networking Conference (CCNC)*. IEEE, 2016, pp. 79–84.

[247] E. K. Wang, Y. Ye, and X. Xu, "Location-Based Distributed Group Key Agreement Scheme for Vehicular Ad Hoc Network," *International Journal of Distributed Sensor Networks*, vol. 2014, pp. 1–8, 2014.

[248] Z. Yan, S. Zeadally, and Y.-J. Park, "A Novel Vehicular Information Network Architecture based on Named Data Networking (NDN)," *IEEE Internet of Things Journal*, vol. 1, no. 6, pp. 525–532, 2014.

[249] M. Chen, D. O. Mau, Y. Zhang, T. Taleb, and V. C. Leung, "VENDNET: VEhicular Named Data NETwork," *Vehicular Communications*, vol. 1, no. 4, pp. 208–213, 2014.

[250] P. Sharma, J. Biddle, A. Daubman, E. Fiore, T. Gallagher, M. Merfeld, S. Paredes, D. Souza, H. Zwahlen, J. Gottschalk *et al.*, "Content and Host-Centric Information Dissemination in Delay-Tolerant Smartphone MANETs ," in *Proceedings of 2012 IEEE Network Operations and Management Symposium*. IEEE, 2012, pp. 586–589.

[251] P. Sharma, D. Souza, E. Fiore, J. Gottschalk, and D. Marquis, "A Case for MANET-Aware Content Centric Networking of Smartphones," in *Proceedings of 2012 IEEE International Symposium on a World of Wireless, Mobile and Multimedia Networks (WoWMoM)*. IEEE, 2012, pp. 1–6.

[252] X. Piao, L. Huang, K. Yuan, J. Yuan, and K. Lei, "The Real Implementation of NDN Forwarding Strategy on Android Smartphone," in *Ubiquitous Computing, Electronics & Mobile Communication Conference (UEMCON), IEEE Annual*. IEEE, 2016, pp. 1–6.

[253] M. Amadeo, C. Campolo, and A. Molinaro, "Named Data Networking for Priority-based Content Dissemination in VANETs," in *Proceedings of 2nd International Workshop on Vehicular Networking and Intelligent Transportation Systems (VENITS 2016)*. IEEE, 2016, pp. 1–6.

[254] Y. Zhang, A. Afanasyev, J. Burke, and L. Zhang, "A Survey of Mobility Support in Named Data Networking," in *Proceedings of the third Workshop on Name-Oriented Mobility: Architecture, Algorithms and Applications (NOM2016)*. IEEE, 2016, pp. 1–6.

[255] E. Borgia, R. Bruno, and A. Passarella, "MobCCN: a CCN-compliant Protocol for Data Collection with Opportunistic Contacts in IoT Environments," in *Proceedings of the Eleventh ACM Workshop on Challenged Networks*. ACM, 2016, pp. 63–68.

[256] Y. Rao, H. Luo, D. Gao, H. Zhou, and H. Zhang, "LBMA: A novel Locator Based Mobility Support Approach in Named Data Networking," *China Communications*, vol. 11, no. 4, pp. 111–120, 2014.

[257] Y. Zhang, H. Zhang, and L. Zhang, "Kite: A Mobility Support Scheme for NDN," in *Proceedings of the 1st International Conference on Information-Centric Networking*. ACM, 2014, pp. 179–180.

[258] Y. Rao, D. Gao, and H. Luo, "NLBA: A Novel Provider Mobility Support Approach in Mobile NDN Environment," in *Proceedings of 2014 IEEE 11th Consumer Communications and Networking Conference (CCNC)*. IEEE, 2014, pp. 188–193.

[259] Y. Rao, H. Zhou, D. Gao, H. Luo, and Y. Liu, "Proactive Caching for Enhancing User-Side Mobility Support in Named Data Networking," in *Proceedings of 2013 Seventh International Conference on Innovative Mobile and Internet Services in Ubiquitous Computing (IMIS)*. IEEE, 2013, pp. 37–42.

[260] R. Ravindran, S. Lo, X. Zhang, and G. Wang, "Supporting Seamless Mobility in Named Data Networking," in *Proceedings of 2012 IEEE International Conference on Communications (ICC)*. IEEE, 2012, pp. 5854–5869.

[261] X. Jiang, J. Bi, Y. Wang, P. Lin, and Z. Li, "A Content Provider Mobility Solution of Named Data Networking," in *Proceedings of 2012 20th IEEE International Conference on Network Protocols (ICNP)*. IEEE, 2012, pp. 1–2.

[262] E. Jung, "A Design of Alias Naming Scheme for Namespace in Named Data Networking," in *Proceedings of 2016 18th International Conference on Advanced Communication Technology (ICACT)*. IEEE, 2016, pp. 537–540.

[263] Y. Tan and S. Zhu, "Efficient Name Lookup Scheme Based on Hash and Character Trie in Named Data Networking," in *Proceedings of 2015 12th Web Information System and Application Conference (WISA)*. IEEE, 2015, pp. 130–135.

[264] W. Quan, C. Xu, J. Guan, H. Zhang, and L. A. Grieco, "Scalable Name Lookup with Adaptive Prefix Bloom Filter for Named Data Networking," *IEEE Communications Letters*, vol. 18, no. 1, pp. 102–105, 2014.

[265] A. Ghodsi, T. Koponen, J. Rajahalme, P. Sarolahti, and S. Shenker, "Naming in Content-Oriented Architectures," in *Proceedings of the ACM SIGCOMM Workshop on Information-Centric Networking*. ACM, 2011, pp. 1–6.

[266] A. Afanasyev, P. Mahadevan, I. Moiseenko, E. Uzun, and L. Zhang, "Interest Flooding Attack and Countermeasures in Named Data Networking," in *Proceedings of IFIP Networking Conference, 2013*. IEEE, 2013, pp. 1–9.

[267] N. Ntuli and S. Han, "Detecting Router Cache Snooping in Named Data Networking," in *Proceedings of 2012 International Conference on ICT Convergence (ICTC)*. IEEE, 2012, pp. 714–718.

[268] A. Afanasyev, J. A. Halderman, S. Ruoti, K. Seamons, Y. Yu, D. Zappala, and L. Zhang, "Content-Based Security for the Web," in *Proceedings of the 2016 New Security Paradigms Workshop*. ACM, 2016, pp. 49–60.

[269] B. Hamdane and S. G. El Fatmi, "A Credential and Encryption Based Access Control Solution for Named Data Networking," in *Proceedings of 2015 IFIP/IEEE International Symposium on Integrated Network Management (IM)*. IEEE, 2015, pp. 1234–1237.

[270] Y. Yu, A. Afanasyev, D. Clark, V. Jacobson, L. Zhang *et al.*, "Schematizing Trust in Named Data Networking," in *Proceedings of the 2nd International Conference on Information-Centric Networking*. ACM, 2015, pp. 177–186.

[271] A. Afanasyev, C. Yi, L. Wang, B. Zhang, and L. Zhang, "SNAMP: Secure Namespace Mapping to Scale NDN Forwarding," in *2015 IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS)*. IEEE, 2015, pp. 281–286.

[272] M. Paknezhad and M. Keshtgary, "Security and Privacy Issues of Implementing Cloud Computing on NDN," *Journal of Basic and Applied Scientific Research*, vol. 4, no. 3, pp. 270–279, 2014.

[273] J. Burke, P. Gasti, N. Nathan, and G. Tsudik, "Secure Sensing over Named Data Networking," in *Proceedings of IEEE 13th International Symposium on Network Computing and Applications, NCA 2014*. IEEE, 2014, pp. 175–180.

[274] ——, "Securing Instrumented Environments over Content-Centric Networking: the Case of Lighting Control," in *Proceedings of IEEE INFOCOM 2013 Workshop on Emerging Design Choices in Name-Oriented Networking Securing*. IEEE, 2013, pp. 394–398.

[275] Y. Wang, T. Pan, Z. Mi, H. Dai, X. Guo, T. Zhang, B. Liu, and Q. Dong, "Name-Filter: Achieving Fast Name Lookup with Low Memory Cost Via Applying Two-Stage Bloom Filter," in *INFOCOM, 2013 Proceedings IEEE*. IEEE, 2013, pp. 95–99.

[276] M. Varvello, D. Perino, and J. Esteban, "Caesar: A Content Router for High Speed Forwarding," in *Proceedings of the Second Edition of the ICN Workshop on Information-centric networking*. ACM, 2012, pp. 73–78.

[277] C. Yi, A. Afanasyev, L. Wang, B. Zhang, and L. Zhang, "Adaptive Forwarding in Named Data Networking," *ACM SIGCOMM Computer Communication Review*, vol. 42, no. 3, pp. 62–67, 2012.

[278] J. Garcia-Luna-Aceves and M. Mirzazad-Barijough, "Enabling Correct Interest Forwarding and Retransmissions in a Content Centric Network," in *Proceedings of the Eleventh ACM/IEEE Symposium on Architectures for Networking and Communications Systems*. IEEE Computer Society, 2015, pp. 135–146.

[279] Y. Wang, D. Tai, T. Zhang, J. Lu, B. Xu, H. Dai, and B. Liu, "Greedy Name Lookup for Named Data Networking," *ACM SIGMETRICS Performance Evaluation Review*, vol. 41, no. 1, pp. 359–360, 2013.

[280] L. Kai, Y. Longyu, and W. Jun, "Scalable Control Panel for Media Streaming in NDN," in *Proceedings of the 1st International Conference on Information-Centric Networking*. ACM, 2014, pp. 207–208.

[281] J. Wei, D. Nguyen, J. Garcia-Luna-Aceves, and K. Nichols, "Experience with Collaborative Conferencing Applications in Named-Data Networks," in *Proceedings of 2013 IEEE 10th Consumer Communications and Networking Conference (CCNC)*. IEEE, 2013, pp. 241–246.

[282] K. Shilton and J. A. Koepfler, "Making Space for Values: Communication & Values Levers in a Virtual Team," in *Proceedings of the 6th International Conference on Communities and Technologies*. ACM, 2013, pp. 110–119.

[283] Z. Zhu, C. Bian, A. Afanasyev, V. Jacobson, and L. Zhang, "Chronos: Serverless Multi-user Chat over NDN," University of California, Los Angeles, Technical Report NDN-0008, Tech. Rep., 2012. [Online]. Available: http://named-data.net/techreport/TR008-chronos.pdf

[284] Z. Zhu, S. Wang, X. Yang, V. Jacobson, and L. Zhang, "ACT: Audio Conference Tool over Named Data Networking," in *Proceedings of the ACM SIGCOMM workshop on Information-Centric Networking*. ACM, 2011, pp. 68–73.

[285] H. Xu, Z. Chen, R. Chen, and J. Cao, "Live Streaming with Content Centric Networking," in *Proceedings of 2012 Third International Conference on Networking and Distributed Computing*. IEEE, 2012, pp. 1–5.

[286] S. Kouyoumdjieva, E. A. Yavuz, O. Helgason, L. Pajevic, and G. Karlsson, "Opportunistic Content-Centric Networking: The Conference Case Demo," *Demonstration at IEEE Infocom*, pp. 1–2, 2011.

[287] Y. Zou, W. Liu, Y. Yang, and J. Xu, "Energy-aware Probabilistic Forwarding in Wireless Content-Centric Networks," in *Information and Communication Technology Convergence (ICTC), 2016 International Conference on*. IEEE, 2016, pp. 270–275.

[288] M. Amadeo, C. Campolo, A. Iera, and A. Molinaro, "Information Centric Networking in IoT Scenarios: The Case of a Smart Home," in *Proceedings of 2015 IEEE International Conference on Communications (ICC)*. IEEE, 2015, pp. 648–653.

[289] ——, "Named Data Networking for IoT: An Architectural Perspective," in *2014 European Conference on Networks and Communications (EuCNC)*. IEEE, 2014, pp. 1–5.

[290] S. H. Ahmed, S. H. Bouk, and D. Kim, "RUFS: RobUst Forwarder Selection in Vehicular Content-Centric Networks," *IEEE Communications Letters*, vol. 19, no. 9, pp. 1616–1619, 2015.

[291] G. Grassi, D. Pesavento, G. Pau, L. Zhang, and S. Fdida, "Navigo: Interest Forwarding by Geolocations in Vehicular Named Data Networking," in *Proceedings of IEEE 16th International Symposium on a World of Wireless, Mobile and Multimedia Networks (WoWMoM)*. IEEE, 2015, pp. 1–10.

[292] G. Grassi, D. Pesavento, L. Wang, G. Pau, R. Vuyyuru, R. Wakikawa, and L. Zhang, "ACM HotMobile 2013 Poster: Vehicular Inter-Networking via Named Data," *ACM SIGMOBILE Mobile Computing and Communications Review*, vol. 17, no. 3, pp. 23–24, 2013.

[293] M. Amadeo, C. Campolo, and A. Molinaro, "CRoWN: Content-Centric Networking in Vehicular Ad Hoc Networks," *IEEE Communications Letters*, vol. 16, no. 9, pp. 1380–1383, 2012.

[294] B. Lucienne and C. Amaresh, *DRM: A Design Research Methodology*. Springer Verlag, 2009.

[295] I. Sommerville, *Software Engineering*. Addison-Wesle, 2011.

[296] B. Boehm and W. J. Hansen, "Spiral Development : Experience , Principles , and Refinements," Spiral Development Workshop, Technical report, Tech. Rep., 2000.

[297] R. S. Pressman, *Software Engineering: A Practitioner's Approach*, 7th ed. McGraw-Hill, 2009.

[298] N. M. A. Munassar and A. Govardhan, "A Comparison Between Five Models Of Software Engineering," *International Journal of Computer Science Issues (IJCSI)*, vol. 5, pp. 95–101, 2010.

[299] A. Suki, "Slight-Delay Shaped Variable Bit Rate (SD-SVBR) Technique for Video Transmission," Ph.D. dissertation, Universiti Utara Malaysia, 2011.

[300] S. A. Nor, "Enhancing Datagram Congestion Control Protocol for Efficient Long Delay Link," Ph.D. dissertation, Universiti Utara Malaysia, 2012.

[301] B. Moore, D. Dean, A. Gerber, G. Wagenknecht, and P. Vanderheyden, *Eclipse Development using the Graphical Editing Framework and the Eclipse Modeling Framework*, 1st ed. IBM Redbooks, 2004.

[302] CCNx, "CCNx Interest Message," Nov. 2013, accessed: 18-Sep-2013. [Online]. Available: https://www.ccnx.org/releases/ccnx-0.8.0/doc/technical/InterestMessage.html

[303] O. Balci, "Verification, Validation and Accreditation of Simulation Models," in *Proceedings of the 29th Conference on Winter Simulation*. IEEE Computer Society, 1997, pp. 135–141.

[304] ——, "Validation, Verification, and Testing Techniques Throughout the Life Cycle of a Simulation Study," *Annals of Operations Research*, vol. 53, no. 1, pp. 121–173, 1994.

[305] Balci1998, *Verification, Validation, and Testing*. John Wiley and Sons, 1998, pp. 335–393.

[306] A. Habbal, "TCP SINTOK: Transmission Control Protocol with Delay-Based Loss Detection and Contention Avoidance Mechanisms for Mobile Ad Hoc Networks," Ph.D. dissertation, Universiti Utara Malaysia, 2014.

[307] S. Hassan and M. Kara, "Simulation-based Performance Comparison of TCP-Friendly Congestion Control Protocols," in *Proceedings of the 16th Annual UK Performance Engineering Workshop (UKPEW2000)*, 2000, pp. 199–210.

[308] O. Ghazali, "Scaleable and Smooth TCP-Friendly Receiver-Based Layered Multicast Protocol," Ph.D. dissertation, Universiti Utara Malaysia, 2008.

[309] H. A. Marhoon, M. Mahmuddin, and S. A. Nor, "DCBRP: A Aeterministic Chain-Based Routing Protocol for Wireless Sensor Networks," *SpringerPlus*, vol. 5, no. 1, pp. 1–21, 2016.

[310] G. Carofiglio, M. Gallo, L. Muscariello, and M. Papali, "Multipath Congestion Control in Content-Centric Networks," in *Computer Communications Workshops (INFOCOM WKSHPS), 2013 IEEE Conference on*. IEEE, 2013, pp. 363–368.

[311] S. H. Ahmed, S. H. Bouk, M. A. Yaqub, D. Kim, and M. Gerla, "CONET: Controlled Data Packets Propagation in Vehicular Named Data Networks," in *Consumer Communications and Networking Conference (CCNC), 2016 13th IEEE Annual*. IEEE, 2016, pp. 620–625.

[312] Q. Mamun, "A Qualitative Comparison of Different Logical Topologies for Wireless Sensor Networks," *Sensors*, vol. 12, no. 11, pp. 14 887–14 913, 2012.

[313] S. Braun, M. Monti, M. Sifalakis, and C. Tschudin, "An Empirical Study of Receiver-based AIMD Flow-Control Strategies for CCN," in *Computer Communications and Networks (ICCCN), 2013 22nd International Conference on*. IEEE, 2013, pp. 1–8.

[314] C. Yi, A. Afanasyev, I. Moiseenko, L. Wang, B. Zhang, and L. Zhang, "A Case for Stateful Forwarding Plane," *Computer Communications*, vol. 36, no. 7, pp. 779–791, 2013.

[315] A. Ndikumana and O. C. Hong, "An Efficient Interest Delivery Scheme Using Named-Data Link State Routing Protocol and Interest NACK in NDN," *Korea Information Science Society*, pp. 1191–1194, 2014.

[316] W. You, B. Mathieu, P. Truong, J.-F. Peltier, and G. Simon, "Realistic Storage of Pending Requests in Content-Centric Network Routers," in *Proceedings of 2012 1st IEEE International Conference on Communications in China (ICCC)*. IEEE, 2012, pp. 120–125.

[317] D. Perino and M. Varvello, "A Reality Check for Content Centric Networking," in *Proceedings of the ACM SIGCOMM Workshop on Information-Centric Networking*. ACM, 2011, pp. 44–49.

[318] O. M. D. Al-Momani, "Dynamic Redundancy Forward Error Correction Mechanism for the Enhancement of Internet-Based Video Streaming," Ph.D. dissertation, Universiti Utara Malaysia, 2010.

[319] I. Ud Din, "Flexpop: A Popularity-Based Caching Strategy for Multimedia Applications in Information-Centric Networking," Ph.D. dissertation, Universiti Utara Malaysia, 2016.