PERFORMANCE EVALUATION OF CACHING TECHNIQUES FOR VIDEO ON DEMAND WORKLOAD IN NAMED DATA NETWORK

SADAQ JEBUR TAHER

MASTER OF SCIENCE (INFORMATION TECHNOLOGY)
UNIVERSITI UTARA MALAYSIA
2016

Perakuan Kerja Tesis/Disertasi

(To be substituted with signed document for this page)

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

Penggunaan Internet dalam konteks kontemporari berkembang pesat terutamanya untuk tujuan paparan maklumat. Ini disebabkan oleh kemunculan 'Information-Centric Networking' (ICN) dalam domain yang lebih luas iaitu akademik dan industri. 'Named Data Network (NDN)' adalah hasil daripada ICN. Di samping itu, NDN diutamakan sebagai seni bina untuk trafik video yang lancar diantara permintaan dan penerima video dalam talian. Penyelidikan ringkas ini mengenal pasti isu semasa yang menyebabkan kesesakan di beban kerja 'Video on Demand' (VoD) yang berpunca daripada penyimpanan kerap objek kandungan dalam repositori tempatan, yang membawa kepada masalah penampanan dan kehilangan paket data. Kajian itu akan menilai teknik cache NDN untuk memilih teknik penggantian cache yang lebih sesuai untuk menangani isu-isu kesesakan, dan menilai prestasinya. Untuk demikian, kajian semasa menggunakan proses penyelidikan berdasarkan 'Design Research Methodology' (DRM) dan pendekatan VoD untuk menerangkan aktiviti-aktiviti utama yang menghasilkan peningkatan dalam hasil akhir penyelidikan. Kumpulan data, serta topologi rangkaian Internet2 dan statistik tontonan video telah dikumpulkan dari platform PPTV. Sebanyak 221 pelayan disambungkan kepada rangkaian dari pusat akses yang sama seperti dalam penggunaan sebenar PPTV. Di samping itu, NS3 menganalisa prestasi metrik teknik penggantian caching (LRU, LFU dan FIFO) untuk VoD dalam 'Named Data Network' (NDN) dari segi nisbah hit cache, pemprosesan, dan keputusan beban pelayan dalam hasil yang munasabah yang muncul berkhidmat sebagai pengganti yang berpotensi dengan pelaksanaan topologi semasa Internet2, di mana nod diedarkan secara rawak. Berdasarkan keputusan, teknik LFU memberikan hasil yang lebih baik untuk kesesakan di antara teknik-teknik lain yang dibentangkan. Akhir sekali, kajian ini mendapati bahawa prestasi metric nisbah cache, kendalian, dan beban pelayan untuk LFU menghasilkan kadar kesesakan yang paling rendah dan mencukupi. Justeru, penyelidik membuat kesimpulan bahawa kecekapan teknik penggantian yang berbeza perlu juga disiasat untuk memberikan pandangan serta idea untuk melaksanakan teknik-teknik ini dalam konteks tertentu. Walau bagaimanapun, keputusan ini memenuhi pemahaman semasa untuk teknik penggantian saiz cache yang berbeza. Setelah teknik penggantian yang berbeza diaplikasikan dan diperiksa, ciri-ciri prestasi dan prestasi jangkaan juga didapati merangsang model cache untuk berjalan cepat merentasi pelbagai aplikasi terbenam.

Kata-kunci: Information-Centric Networking (ICN), Named Data Network (NDN), Video on Demand (VoD), Congestion, Caching Replacement Techniques

Abstract

The rapid growing use of the Internet in the contemporary context is mainly for content distribution. This is derived primarily due to the emergence of Information-Centric Networking (ICN) in the wider domains of academia and industry. Named Data Network (NDN) is one of ICN architectures. In addition, the NDN has been emphasized as the video traffic architecture that ensures smooth communication between the request and receiver of online video. The concise research problem of the current study is the issue of congestion in Video on Demand (VoD) workload caused by frequent storing of signed content object in the local repositories, which leads to buffering problems and data packet loss. The study will assess the NDN cache techniques to select the preferable cache replacement technique suitable for dealing with the congestion issues, and evaluate its performance. To do that, the current study adopts a research process based on the Design Research Methodology (DRM) and VoD approach in order to explain the main activities that produced an increase in the expected findings at the end of the activities or research. Datasets, as well as Internet2 network topology and the statistics of video views were gathered from the PPTV platform. Actually, a total of 221 servers is connected to the network from the same access points as in the real deployment of PPTV. In addition, an NS3 analysis the performance metrics of caching replacement technique (LRU, LFU, and FIFO) for VoD in Named Data Network (NDN) in terms of cache hit ratio, throughput, and server load results in reasonable outcomes that appears to serve as a potential replacement with the current implementation of the Internet2 topology, where nodes are distributed randomly. Based on the results, LFU technique gives the preferable result for congestion from among the presented techniques. Finally, the research finds that the performance metrics of cache hit ratio, throughput, and server load for the LFU that produces the lowest congestion rate which is sufficient. Therefore, the researcher concluded that the efficiency of the different replacement techniques needs to be well investigated in order to provide the insights necessary to implement these techniques in certain context. However, this result enriches the current understanding of replacement techniques in handling different cache sizes. After having addressed the different replacement techniques and examined their performances, the performance characteristics along with their expected performance were also found to stimulate a cache model for providing a relatively fast running time of across a broad range of embedded applications.

Keywords: Information-Centric Networking (ICN), Named Data Network (NDN), Video on Demand (VoD), Congestion, Caching Replacement Techniques

Acknowledgements

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

"*Work; so Allah will see your work and (so will) His Messenger and the believers*;"

(The Holy Quran - AtTawbah 9:105)

Conducting this research marks the end of an interesting and eventful journey. The completion of this dissertation signifies the fulfillment of a long-awaited goal. It could not have been achieved without the professional academic and personal support of the following wonderful and talented people.

I will start by extending my deep and sincere gratitude to my supervisors Dr. Mohammed K.M. Madi and Professor Dr. Suhaidi Hassan (School of Computing, Universiti Utara Malaysia) for their tireless encouragement, wisdom and experience, who provided me with constant guidance and constructive criticism throughout all stages of my research. I must say a huge thank you to the current and past members of InterNetWorks Research Lab whom I enjoyed working with; especially, Dr. Massudi Mahmuddin, Dr. Nur Haryani Zakaria, Dr. Rohaida Romli, Dr. Ahmad Suki Che Mohamed Arif, Dr. Adib Habbal, Dr. Mohd. Hasbullah Omar, Dr. Norliza Katuk and others.

Finally, my heartiest gratitude goes to my family, to my father and mother lates , to my beloved wife (MSc. Rasha) for her understanding, support, and love, also deepest thanks to my sons (Jaafer and Osamah), and my daughter (Tabarak) for being so sweet and loving, to my wife's mother and my sister who always has faith in me and prays for my success,to my soul (Tareq) , to my brothers , to my wife's brothers, and to my friends who are willing to extend a helping hand.

Table of Contents

List of Tables

List of Figures

List of Abbreviations

ABR	-	Adaptive-Bit-Rate
ADSL	-	Asymmetric Digital Subscriber Line
CDN	-	Content Delivery Network
CS	-	Content Store
DoS	-	Denial-of-Service attack
FIB	-	Forwarding Information Base
FIFO	-	First-In-First-Out police
ID	-	IDentifier
IP	-	Internet Protocol
IPTV	-	Internet Protocol Television
ISP	-	Internet Service Provider
LFU	-	Least Frequently Used
LRU	-	Least Recently Used
NDN	-	Named Data Networking
NLR	-	National Lambda Rail
NS3	-	Network Simulation version 3
PC	-	Personal Computer
PIT	-	Pending Interest Table
PPTV	-	Platform Provider Television
P2P	-	Peer-to-Peer
TCP	-	Transmission Control Protocol
UCLA	-	University of California, Los Angeles
URL	-	Uniform Resource Locator
VoD	-	Video-on-Demand
TCP/IP	-	Transmission Control Protocol/Internet Protocol

CHAPTER ONE
INTRODUCTION

This chapter provides an overview of this study. It includes a brief introduction of Named Data Networking (NDN) technique and its application such as Video-on-Demand. Besides, the chapter contains the research problem and the research questions which are in line with the research objectives. The scope and significance of this research is also explained in this chapter.

1.1 Background of the Study

Named Data Networking (NDN) has been identified as the communication architecture used in an Internet video storage. It gives high priority to the data which include videos. The NDN architecture is basically comprised of communication units as shown in Figure 1.1 as interest packet and data packet [1].

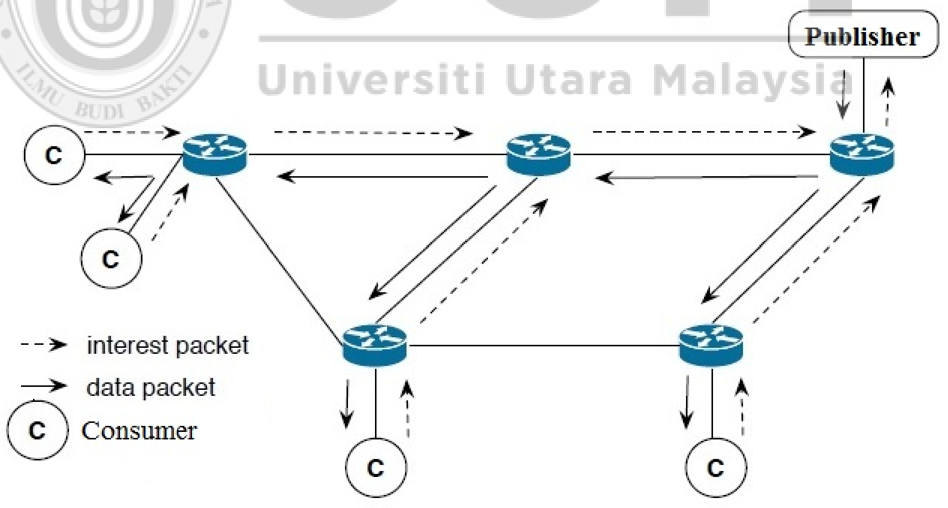


Figure 1.1: Data Flow in NDN Architecture [1]

The interest packet is one of the communication units in NDN sent upon the request of consumers of data or video steaming, while the data packet is protected by a cryptographic signature giving most integrity of the data video source from the cache.

Therefore, the interest packet carries a selector field to indicate instances in case there are multiple data to satisfy the interest packet [1, 6]. On the other hand, the data packet, basically satisfies the interest packet since the interest packet has carried the prefix name of the data [6].

In addition, the notable Internet architecture, such as TCP/IP, provides IP addresses to the host and thus, projects Internet as a point-to-point communication system [1]. NDN has been referred to as the Internet architecture that provides benefits over TCP/IP due to its ability to give a venue to content distribution (such as video streaming) to reach the designated destination successfully [2]. NDN ensures that consumers retrieve data content through the use of specified hierarchical data names [1, 2] as shown in Figure 1.2. This implies that NDN architecture is capable of representing future Internet architecture so as to ensure accuracy and prompt delivery of online content distribution.

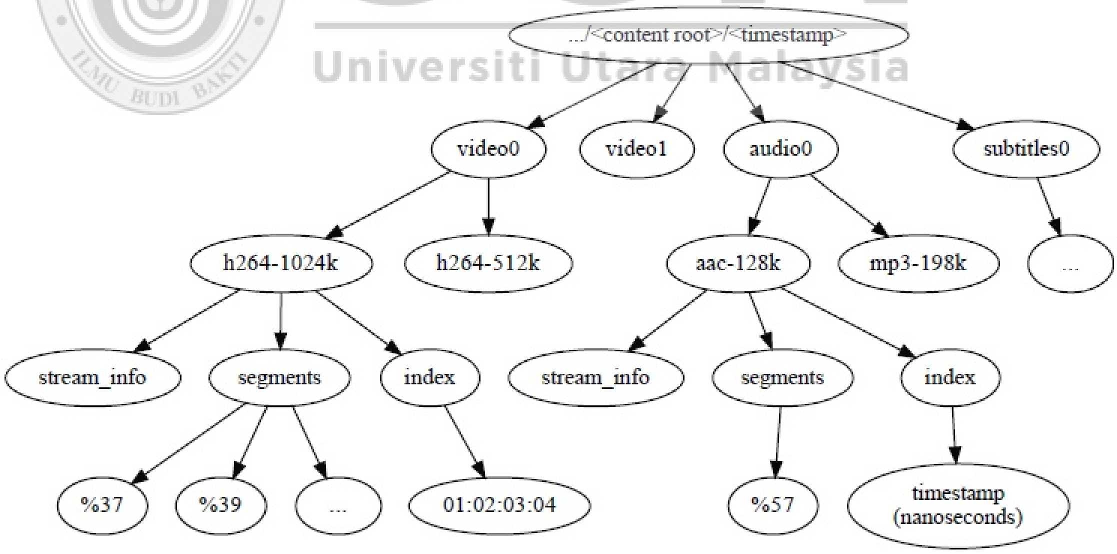


Figure 1.2: Video Content Streaming with Specified Hierarchical Data Name [2]

Furthermore, the NDN-video protocol is comprised of two participants, publisher and consumer, while the streaming processes are through one-to-many [2]. The NDN-video architecture ensures that a publisher publishes or releases, data which is received or used by many consumers, thus, leading to the workload of the cache or storage. Besides, the operation of NDN architecture in the cache provides the data consumer's ability to release interest packets or request that bears an assigned name of the expected contents [1, 2].

Over the recent years, Video-on-Demand (VoD) has become a rapidly popular service in the Internet. However, it is an expensive service due to its high bandwidth requirements and high number of requests. Internet on-demand video traffic has been rapidly growing while on-demand video streaming or VoD has become extremely popular in the Internet. For example, YouTube is a video-sharing service that streams its videos to users on demand with about 20 million views per day and a total viewing time of over 10,000 years to date. Other examples of major Internet VoD publishers are MSN Video, Yahoo video, Google Video, and CNN. Most of the VoD services streamed over the Internet is encoded in the range of 200-400 kbps for which video publishers are typically charged at 0.1 to 1.0 cents per video minute by ISPs (or CDNs) [7]. YouTube itself pays over 1 million dollars a month for its bandwidth expenses, whereas MSN VoD has a billing rate corresponding to 2.20 Gbps [8]. With the explosive increase of higher quality video demands up to 3 Mbps or more, these costs are expected to further go up. This causes considerably serious challenges of deploying VoD services at large scale as mentioned below [9]:

• The enormity of the video catalog size makes a content providers' effort to replicate the video library in different locations totally wasteful and non-scalable.

• Higher video quality requires efficient utilization of network bandwidth resources which is very expensive.

• The emergence of various device modalities demands design efficiency in systems and techniques in order to cater to the requirements of users as well as device features.

1.2 Motivation

The rapidly growing use of the Internet in the contemporary context is mainly for content distribution. This is derived primarily due to the emergence of Information-Centric Networking (ICN) in the wider domains of academia and industry [10]. Its use can be traced back in the 1960s and 1970s, when point-to-point paradigm of the Internet was widely used in the beginning of computer networks. In the current context, the Internet is mainly dominated by rich multimedia content. Being the forerunner of the future Internet, ICN proposes a redesign of the network infrastructure built around the content which is considered as a notable paradigm shift from the current sender-driven point-to-point communication to a receiver-driven content distribution communication [11]. This new network architecture aims at resolving many issues of the current Internet (such as content distribution efficiency, security, congestion, etc.) and thus providing a better network communication to all.

The main platforms used for the Internet or online video streaming are linked to web browser plug-in, game console, and TV. It is supported by the video streaming technique between providers and clients. Studies have found out that videos are sourced from either the Internet or traditional way which requires completion before engaging in playback, thus both ways are characterized with the traffic issue [12]. Previous studies have stressed that many of the Internet videos are encoded at different rates ranging from 235Kb/secs standard definition to 5Mb/secs high definition [13, 14]. This empowers the Internet videos to be stored as separate files on the server and cause more work for the storage cache. Sometimes, large storage by the Internet video streaming experiences failure while users access it. This happens due to the workload (number of different yet related entities) of VoD. VoD is a system which allows users to select and watch video content when they choose to, rather than having to watch at a specific broadcast time. IPTV technology is often used to bring VoD to televisions and personal computers [6].

1.3 Problem Statement

The NDN has been emphasized as the video traffic architecture that ensures smooth communication between the request and receiver of online video [15]. Therefore, it enables the cache storage to attend to several forms of request of online videos simultaneously [6, 15].

In addition, researchers have stressed that the NDN is flexible and supports many of encoding formats allowed by streamer which is a pipeline-based multimedia framework [2]. Thus, it allows the publishers to obtain a signed content object from the input video stream that could either be live or pre-recorded and stored in local repositories. However, immediate and frequently storing of signed content object into the local repository has caused congestion of VoD due to the persistent workload [1, 16, 17].

Furthermore, frequent congestion in VoD workload leads to buffering of request content, especially in the Internet video request [17]. Ideally, handling of large data of video is controlled by the inner layer which prompts the information needed to allow the compatibility of NDN-video with frame rate codes and ensure buffer playback. But multiple buffering of request videos leads to data packet loss due to the congestion in the cache storage network [1, 2, 15, 16, 17]. Therefore, it is important to further improve the architecture used in Internet video streaming in order to deal with the congestion issues that are further associated with buffering issues and data packet loss.

The concise research problem of the current study is the issue of congestion in VoD workload [1, 2, 16, 17] caused by frequent storing of signed content object in the local repositories, which leads to buffering problems and data packet loss. The study will assess the NDN cache techniques to select the best cache replacement technique suitable for dealing with the congestion issues, and evaluate its performance.

1.4 Research Questions

The completeness of the current study will produce answers to the following research questions:

i. What is the preferable NDN cache replacement technique that is suitable for improving congestion during VoD workload?

ii. How can evaluate the performance of the replacement techniques for the NDN cache to improve congestion during VoD workload?

1.5 Research Objectives

The objectives of the current study are:

1. To determine the preferable cache replacement technique that is suitable for improving congestion during VoD workload by assessing existing NDN cache replacement techniques.

2. To evaluate the performance of the cache replacement technique of NDN during VoD workload in terms of throughput, cache hit ratio and server load.

1.6 Research Scope

This study is basically towards assessing the congestion in the NDN cache technique used in the VoD workload. Studies have shown that viewership of video over the Internet is increasing exponentially, while the content distribution of video data has been distressed due to the data congestion [2, 15, 18]. Thus, it is required to assess the NDN cache technique in order to stop the current long time buffering of data request of Internet video caused by data congestion in the repository. The scope of research for the current study is illustrated in Figure 1.3 below.

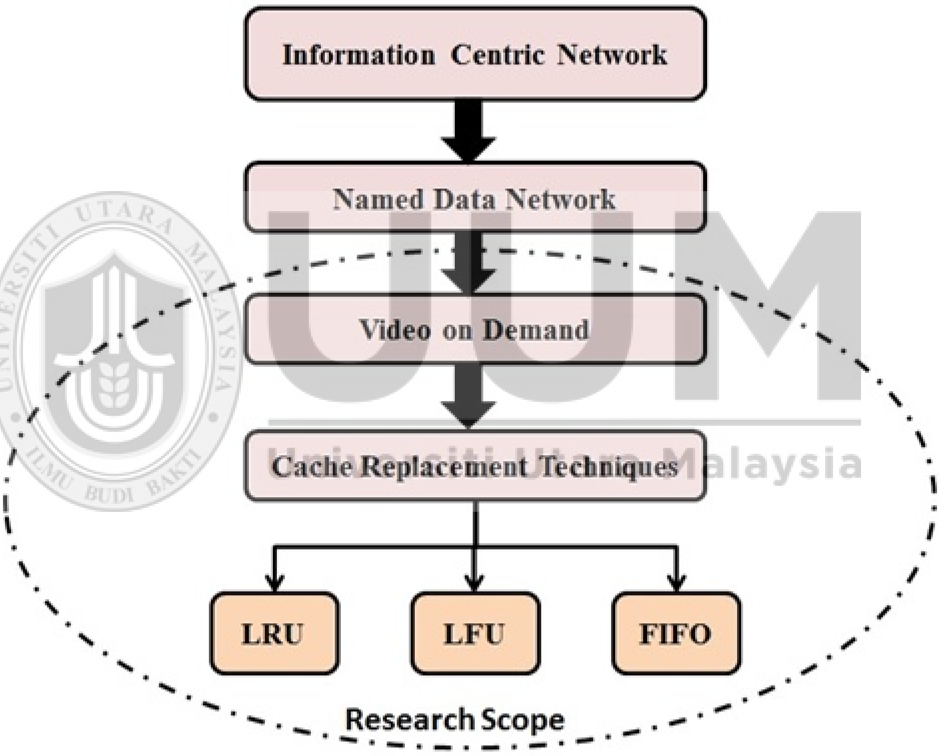


Figure 1.3: Scope of Research

As shown in Figure 1.3, the current study provides a broad outline of Information-Centric Networking (ICN) and more specifically, the Named Data Networking (NDN) technique. NDN is a communication architecture used for Internet video storage. Due to video streaming happening on a large scale, the NDN architecture frequently faces congestion issues during VoD workload which lead to buffering of the on-demand

Internet video requests. This further demands assessment of the NDN to sort out the congestion problem by reducing it. The current study will assess the NDN architecture to identify the causes of congestion. Relevant cache replacement techniques, such as LRU, LFU, and FIFO, will be used to assess the NDN cache technique by executing with four cache size (1GB, 10GB, 100GB, 1TB) in order to ensure eradication or reduction of congestion during VoD workload. These cache replacement techniques will be evaluated in terms of key performance metrics in order to determine the best replacement techniques for the NDN cache technique to resolve the congestion issue of VoD workload.

1.7 Significance of the Research

This study would have a practical contribution to the demand for more Internet videos once the repository is lower of data congestion. Thus, the benefit of VoD workload would affect both users and providers of Internet videos positively [6]. In other words, there would be more revenue for the Internet service providers and indirectly the host of online video once the long and frequent buffering of the request of VoD workload is reduced, if not completely stopped. Moreover, the study would assist academically with integrating the previous works on NDN, online video and Internet of Thing.

1.8 Dissertation Outline

Chapter One of the dissertation provides the introductory ideas of what the study is all about. It contains a brief outline of the research, including the problem statement, research questions and objectives, and the scope and significance of the research. Chapter Two provides an extensive review of previous and related literature and focus on the online video streaming and NDN video streaming architecture. Chapter Three focuses on the designing of methodological approach to achieve the objectives of this research. Chapter Four presents the results on the implementations of the three cache replacement techniques using the NS3 simulator. Chapter Five contains the discussions of the results. Finally, Chapter Six has the conclusion and future works.

CHAPTER TWO
LITERATURE REVIEW

This chapter discusses the relevant literature that is related to the context of this study. Most of the literature that is highlighted in this chapter are part of the requirement for achieving the objectives of this study. It contains a brief overview of ICN network. The chapter mainly focuses on the overview of the NDN, its architecture, and attributes. The chapter also discusses the issues in the video on demand, cache replacement techniques and the causes of congestion.

2.1 Information-Centric Networking

Information-Centric Networking (ICN) is a new networking archetype which focuses on skeptical location-based content identifiers rather than host identifiers. ICN has been implemented to propose several network architectures. One of the most important characteristics of ICN is the in-network caching capability, which is enabled by the skeptical location-based content identifiers and explicit naming of contents. However, there is still a shortage of suitable public tools for effective performance evaluation of caching systems. The available simulators are either bound to a specific architecture or cannot simulate at the required scale within the desired time-frame. These problems can be addressed by Icarus, which is a Python-based caching simulator for ICN [19]. Icarus provides modeling tools for caching research and is used for evaluating ICN-implemented caching techniques.

In-network caching is significantly in content-driven Internet which is based on ICN frameworks [11]. ICN can deliver a named block of data anywhere. It contains content-oriented data chunks [20] which are atomic units explicitly addressed by name [21] (Such as \objects", \items", and \chunks") and which do not differ independent data items from segments [22]. ICN conducts systematic caching without any tedious processing tasks such as HTTP header parsing. Universal in nature [11], ICN caching contains Content Distribution Networks [23], which is a subset of paying content providers, as well as Internet users of all sizes or status with their content cached. However, this universal nature of ICN leads to severe security concerns [24, 25, 26] such as agility of denying service attacks by injecting corrupted content in network caches [27]. The digital signature associated with every chunk [28, 29] permits each ICN node to reliability verify content and source. The valid objects are then identified and stored by network caching.

2.2 Overview of NDN

The Named Data Network or NDN has been widely studied by researchers over recent years, which it is one of ICN architectures [10]. Studies have shown that NDN has been suggested as the evolution of IP architecture to circulate thin waist role [30, 3]. Figure 2.1 shows the packet-based NDN, which can be renamed and other objects from the endpoint communication. NDN can change the semantics of the network service delivery package to a specific destination address to get the data by identifying a specific name. Thus, the design and functionality of the NDN architecture makes it easier for the user's choice and competition with the network evolution [31, 32].

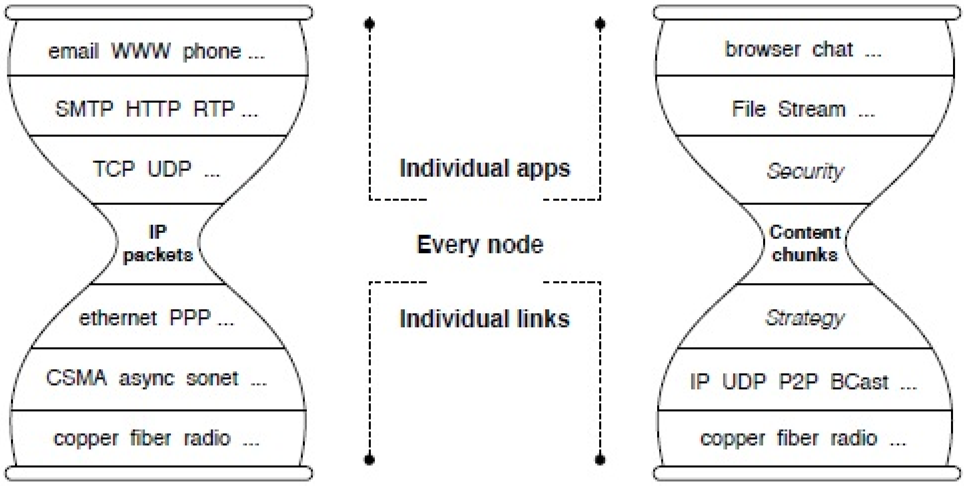


Figure 2.1: Building blocks of the NDN Architecture [3]

The name in an NDN packet is capable of naming objects like end point, a piece of data into a movie or a book and ordered to turn on some lights. It can change the concept of the NDN networks towards the use of the Internet that have been tested physically. However, the NDN architecture is unable to resolve a much wider range of challenges, including end-to-end communications group, distribution problems, and content control. Nonetheless, the NDN along with the strengths and weaknesses of the current Internet structure builds in the design primitives security and self-regulation of network traffic [33]. Moreover, NDN architecture has been described by researchers as the outcome of the previous instance of the general direction of research network that emerged with different architectural designs [10, 34].

2.2.1 NDN Architecture

Communication and motivation in the NDN by the receivers, which are referred to as consumer data, happens through the exchange of two types of package: interest and data [4]. Each package bears a name that identifies a piece of data that can be transmitted in a single data packet [34]. The consumer function primarily puts the name of the desired piece of data in the interest packet and sends it to the network. On the other hand, routers use a unique name for the assignment of interest about the data product. Studies have revealed that once interest rates reach the node containing the required data, the node returns the data packet containing both the name and the content [10, 35]. Besides, NDN architecture has the signing key used for the product, which connects the two beams as shown in Figure 2.2. Thus, the trace in the reverse path makes the interest package to return to the consumer requesting a packet data.

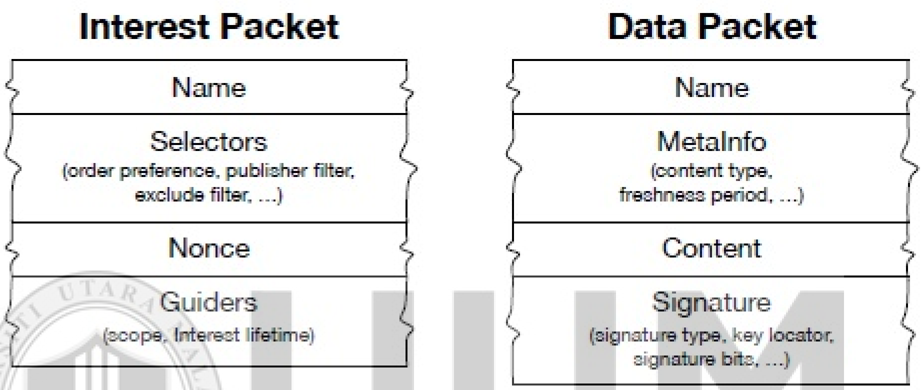


Figure 2.2: Packets in the NDN architecture [4]

2.2.2 Operation of NDN Architecture

The data and interest packets can be forwarded with each router of NDN maintaining three data structures: Pending Interest Table (PIT), Forwarding Information Base (FIB), and Content Store (CS) as shown in Figure 2.3. In addition, the operation is performed from the re-routing strategy that determines whether, when and where to redirect each tray interest in NDN architecture. All the stores of PIT Interests referred to a router, but not satisfy the stage [36, 37, 38].

Moreover, the interest package arrived through a router NDN leads to a preliminary examination of the content store to match the data, thus restoring the data packets on the router interface and the interest packets on their sources. Else, the router is used to search for the name of the PIT. In case of any matching entry, it simply records the incoming interface of interest in the introduction of PIT [37, 5]. If the input matching PIT is absent, it can cause the router to redirect attention to the data product which is based on information in the FIB along with adaptive forwarding router strategy [10, 39]. Thus, guiding Interests of the same name have multiple downstream of the contract and forward only the initial data source about the product. Thus, the FIB is characterized with the prefix-based routing protocol name, and can have multiple interface output for each prefix [39].

In addition, the redirection in the NDN sometimes leads to dropping interest in a strategy whenever the upstream links are congested or suspected of interest to be part of a denial of service attack [40]. Moreover, all the interests in the package ensure that the re-routing strategy retrieves the longest prefix, links it with the FIB input, and decides the referral points and the time of the interest. Content Store of NDN is a temporary cache for data packets received by the router. In other words, the independent feature of the package NDN data is meaningful whenever it has been temporarily stored to meet the interest in the future [10, 39].

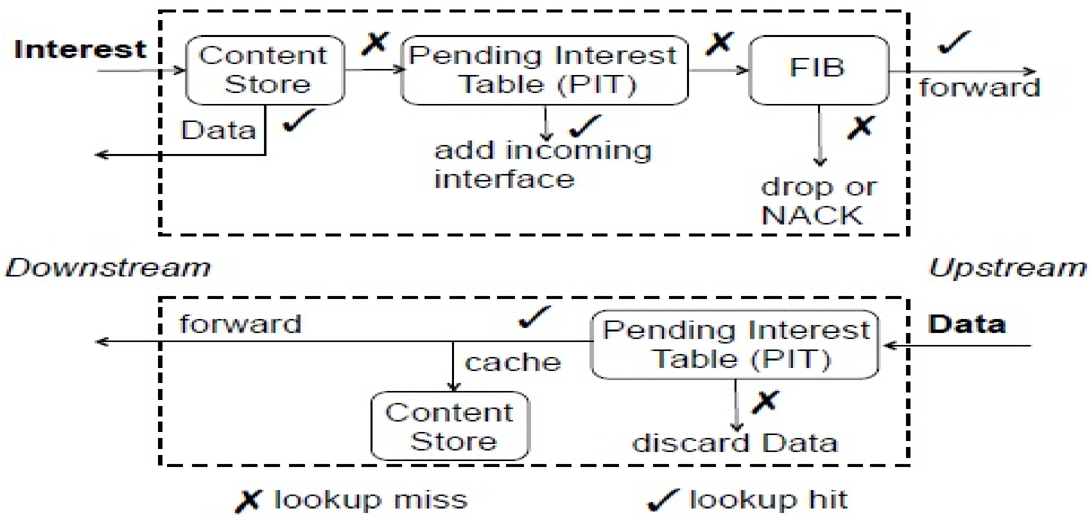


Figure 2.3: Forward processing in NDN Node [5]

The arrival of the packet data helps NDN routing device to find the PIT entry matching and routing data to all downstream interfaces included in the entry PIT, and thus the entry is removed from PIT and the data is stored in the content store. At the same time, data packets always take the reverse path of interest, results, and one benefit package in a single data packet on each link in the absence of packet losses. Moreover, no data packets carry any host or interface addresses. Routers route interest packets towards data producers [41]. Based on the names carried in the packets, forward data packets are ensured based on the information PIT set up by the state to its interests at every hop of consumers [14, 33]. Thus, the symmetry of interest / data package pays leap after leap loop control, and eliminates the need for any idea of the source or destination nodes in data delivery, contrary to what happened in the IP model for packet delivery end-to-end exchange.

2.2.3 Attributes of NDN

2.2.3.1 Routing and Forwarding

NDN forwards and routes packets to the names which eliminate problems caused headlines in architecture such as the IP address space exhaustion, NAT traversal, and management of the address. Although there is no problem of exhaustion title since the namespace is unlimited, there is a problem NAT traversal in existence since NDN eliminates public and private addresses. Generally address assignment and management is not required in local networks. Thus, the NDN is able to use the traditional routing techniques such as link-state and distance vector rather than the announcement of IP prefixes whose names and covers are announced by router NDN and are directed to serve the intended data [30, 3]. Thus, the routing protocol propagates these ads across the network, and informs all construction-oriented networks with their own FIB.

The PIT state in each router supports re-routing through the plane NDN data, and recording all outstanding interfaces containing benefits and the removal of interest after receiving the matching data or timeouts. Based on the information in the FIB and performance measurements, unit redirection to adapt in each router makes informed decisions about Interests forward strategy which interfaces [3, 5]. Besides, it also focuses on the number of unsaturated interests to allow the PIT, the relative priority of different interests, disclosure of interest routing between multiple interfaces, and choose alternative routes to avoid the failure load balancing [38, 5].

2.2.3.2 In-network Storage

NDN package bearing the name and signature of the data includes requests or retrieved independent projects. Thus, the router can cache receiving data packets in the content store, and use it to meet future demands. Besides, the content store is similar to the buffer memory in IP routers, but IP routers cannot be reused after the package is forwarded to the destination, while the NDN routing devices can. In some cases, NDN treats storage channels and a similar network in terms of data retrieval. In statistical context and content delivery, NDN achieves nearly optimal data delivery and takes advantage of caching in the case of multicast or retransmission after packet loss [3, 5].

Furthermore, caching data raise consumer name different from that IP, which allows examination of packet headers' privacy concerns and loads data consumers [3, 34]. Naming and caching of data in NDN networks does not facilitate control what data is requested, but without the destination addresses it is difficult to determine who requests it. Therefore, the NDN offers a radically different kind of privacy protection of existing IP networks. Many caching researchers have confirmed in the network, such as the basic gains of buildings ICN in context [34]. Despite the fact that the NDN can support more powerful CDN architectures of TCP / IP, NDN provides many other functions such as data leading to a number of insurance gains, and shows a large and important advantages [30, 3].

2.2.4 Congestion in NDN

Extensive research has been conducted on congestion handler in NDN. Most of the previous perform follow the actual end-to-end technique of TCP through adjusting one Interest to suit a certain network characteristics and usage settings [31]. However, the heterogeneous Round Trip Time (RTT) due to caching, multiple providers and multipath forwarding makes it hard to line retransmission timeout as in TCP is designed to provide further support to detect blockage [32]. Consequently, some solutions seek to help predict RTT a lot more accurately through certain areas based on the association of stored data with future content relevant to the data packets.

Congestion control in NDN is always recognized to be a smart and adaptive way in which the information is processed similar to the IP manner [3]. In addition, the congestion control of NDN also consists of routing, which serves a similar purpose as in an IP network for the aim of computing direction-finding tables through which the process for forwarding information in an NDN network is split into a two-step procedure: client first sends interest packets, then info relevant to that interest packets flow back on the same path inside the reverse direction [30].

On the other hand, a number of perspectives can be attained when dealing with congestion enhancement in NDN based routers. This includes balancing the pending requests associated with clients' interest to steer Data packets time for requesting packets. The impending Interest condition, together with all the two-way info makes it possible for NDN routers' forwarding procedure to measure performance associated with different trails. It also includes determining any potential failures as well as retry alternative paths. Such features associated with NDN is referred to as adaptive forwarding that provides the possibility for routers to detect network issues quickly without anticipating global direction-finding convergence. However, in the event that data is not received before the timer expires, there might be potential forwarding issues where NDN routers can begin exploring alternative paths instantly using regional state info [3].

The moment congestion arises; data retransmission is aided by simply caching the attached information within the same path. For instance, in the event that more than one congested inbound link is found over the path between the client and server in which the data supply gets through the first infected link may not make it in the second link [39]. After this, caching as a result will permit the information dedicated to certain packet to be retransmitted more than only the other congested link. Meanwhile, the current characteristics of Internet make it difficult to control the congestion in NDN where the retransmission connected with data could happen all the way backs to the server. This can lead to increase the number of trails packets need to pass a certain link [5]. As such, it can be concluded that NDN help to make steady progress towards the final destination for the aim of fulfilling the clients' Interest together with retransmitted packets. In addition, NDN reduces the risk of congestion collapse, which could occur throughout today's Internet when bandwidth simply repeated retransmissions.

2.2.5 Video on Demand

Replacement technique defines the enhancement factor in deciding the system performance [42]. Different policies have different effects on system performance, but there is a performance gap between processors and memory system. For instance, IPTV services provides various services to the user as per their request. The proposed technique is based on the total number of hit and miss counts on the files on every access, whereas replacement is based on the most recently accessed counts, which perform better than other replacement policies like LRU, LFU, and FIFO etc.

Previous studies have revealed that the video represents 66% of total Internet traffic and accounts for more than 40% of the cellular traffic [33, 43]. This demand cellular providers have led to massive deployment capabilities to support high-quality streaming video [33]. In spite of these efforts, reliable video over cellular networks has proven to be difficult. In some cases, it was found that part of the stalled videos with the video quality increases with 10.5% of the videos 240p stall while 45.7% of the videos (720P) suffer from procrastination [44].

Most of the content providers today use Adaptive-Bit-Rate (ABR) streaming where the goal is to match the parts of the video delivery rate to the bandwidth available end-to-end [14, 33]. In other words, the principle of a simple and practical application has to infer the available bandwidth and adjust the price cut while balancing metrics such as quality, and cuts the number of rate switches [3, 33]. In addition, the state-of-the-art techniques do not comply with how to infer the available and historical bandwidth productivity usage, while others use the buffer occupancy traffic shaping [36, 39].

Moreover, reasoning minutes of display available is non-trivial because of varying bandwidth capabilities, link congestion, and other factors [3]. This is a difficult task, especially in the cellular networks due to the inherent variability in signal strength, interference, noise, and movement causing the user bandwidth to vary widely over time [36]. With the correct result, there is not a great match between the estimates used by the existing techniques, and the bandwidth of the network actually available, leading to a poor quality of experience over cellular networks [36, 39]. On the other hand, recent studies have opened the possibility of a promising technique in order to accurately predict the available short and medium time scale bandwidth [32, 35].

2.2.6 Congestion in VoD

Congestion plays important role in determining performance of any networks. With more congestion, performance of networks deteriorates. In the context of the congestion issues in VoD, it is mainly caused by frequent storing of videos requested in the local repositories. In case of a pipeline-based multimedia framework, publishers can obtain the signed content object from the input video stream which can be live or pre-recorded, and stored in the local repository [2]. As online videos are requested frequently, the signed content objects are stored quite frequently into the local repositories. This further leads to congestion of videos on demand due to the continuous workload [1, 16, 17]. Moreover, congestion in the workload leads to buffering of the requested videos. But multiple buffering of such video requests can cause loss of data packet due to congestion in the storage network.

2.2.7 Cache Replacement Technique

Cache replacement technique plays a crucial role in the web caching. Replacement techniques are necessary to achieve the high sophisticated cache mechanism. The replacement technique is helpful in the eviction of the object from the cache and building new space for the incoming object. Due to its limited size, a cache cannot store the entire requested object [45]. So, the cache replacement technique provides room for new documents [45, 46]. This is applicable when the cache is full of objects and the new object is then inserted into the cache. There are several replacement policies in caching techniques which are presented below.

The reasearcher chooses three cache replacement techniques which they are Least Recently Used (LRU), Least Frequently Used (LFU), and First-In-First-Out (FIFO), because they are the most common video replacement techniques and they are used widely in today's network [47]. Most of these techniques do not conform to the principle of locality according to which the memory references tend to be clustered over a short period of time. Thus, by this principle, if a video has been referenced, there are chances of its future reference as well. For instance, First-in-first-out (FIFO) technique tries to give reference to those videos which have been referenced once.

2.2.7.1 Least Recently Used (LRU)

The LRU video replacement technique is simple and easy to use. This technique works on the timestamp. It removes the least recently used object that was not used for a long time when the cache exceeds its maximum size and puts a new object in place of the evicted object [45, 48, 49]. In case of the cache size is not full, it will insert the object in the cache memory.

Table 2.1: Example of LRU Video Replacement Technique

Requests	*V7*	*V0*	*V1*	*V2*	*V0*	*V3*	*V0*	*V4*	*V2*	*V3*
Part 3			*V1*	*V1*	*V1*	*V3*	*V3*	*V3*	*V2*	*V2*
Part 2		*V0*	*V0*	*V0*	*V0*	*V0*	*V0*	*V0*	*V0*	*V3*
Part 1	*V7*	*V7*	*V7*	*V2*	*V2*	*V2*	*V2*	*V4*	*V4*	*V4*

In addition, in order to clearly understand LRU technique, the researcher assumes having "10" requests as video pages, that are arranged as (*V7, V0, V1, V2, V0, V3, V0, V4, V2*, and *V3*) sequentially based on the time of the request the object as shown in the above Table 2.1 that presents an example of how LRU caching replacement technique works. Also, we suggest that the three parts of cache memory are empty when starting, so for example, the first three requests (*V7, V0, V1*) are inserted in the three parts of cache memory (Part 1, Part 2, and Part 3) as sequentially because these three parts of cache memory are empty.

Furthermore, the request (V2) is saved instead of the request (*V7*), because of two reasons: (1) The three parts of cache memory are saved three requests of video (*V7, V0, V1*), that makes the cache memory full, (2) the request (*V7*) is the least recently used (In other words, *V7* is the request that was done before the requests *V0*, and *V1*).

2.2.7.2 Least Frequently Used (LFU)

The LFU video replacement technique removes videos which are least frequently used and then put a new requested video in free part of cache memory [45, 46, 48, 49]. It is very simple and easy to use. In this, a counter which will count the frequency of the video is maintained. The video which is less frequent to use is replaced by the new incoming video.

Table 2.2: Example of LFU Video Replacement Technique

Requests	*V7*	*V0*	*V1*	*V2*	*V0*	*V3*	*V0*	*V4*	*V2*	*V3*
Part 3			*V1,1*	*V1,1*	*V1,1*	*V3,1*	*V3,1*	*V3,1*	*V2,1*	*V2,1*
Part 2		*V0,1*	*V0,1*	*V0,1*	*V0,2*	*V0,2*	*V0,3*	*V0,3*	*V0,3*	*V0,3*
Part 1	*V7,1*	*V7,1*	*V7,1*	*V2,1*	*V2,1*	*V2,1*	*V2,1*	*V4,1*	*V4,1*	*V3,1*

In addition, in order to clearly understand LFU technique, the researcher assumes having "10" requests as video pages, that are arranged as (*V7, V0, V1, V2, V0, V3, V0, V4, V2*, and *V3*) sequentially based on the time of the request the object as shown in the above Table 2.2 that presents an example of how LFU caching replacement technique works. Also, we suggest that the three parts of cache memory are empty when starting, and the frequency count has an initial value begin with zero to all requests (Frequency of all requests=0). So for example, the first three requests (*V7, V0, V1*) are inserted in the three parts of cache memory (Part 1, Part 2, and Part 3) sequentially because these three parts of cache memory are empty.

While, the request (*V2*) is saved instead of the request (*V7*), because of three reasons:

(1) The three parts of cache memory save three requests of video (*V7, V0, V1*), that makes the cache memory full, (2) all the three parts of cache memory have the same value of the counting of frequent (Frequence=1), (3) the request (*V7*) is the request that was done before the requests (*V0*, and *V1*).

Moreover, the last request (*V3*) gives another case that saves the request (*V3*) instead of the request (*V4*). This case is done because of three reasons: (1) the cache memory is becoming full, (2) the counting of frequent is different (Frequency of *V4*=1, Frequency of *V0*=3, and Frequency of *V2*=1), therefore the request (*V0*) must not replace with it another request because it has a high frequency, (3) the request (*V4*) is the request that was done before the request (*V2*).

Since a network proxy is required to serve thousands of requests per second, the overhead needed to do so should be kept to a minimum. To that effect, those Videos that are less frequently used should be evicted. Hence, the frequently used videos are retained at the expense of the less frequently used, since the former have proved themselves to be useful over a period of time [46]. Many arguments have been placed to counter this assertion that the frequently used videos may not be needed in the future, however, this is not a usual case in most situations. For instance, the static videos of heavily used videos are often requested by every user of that video. While using the LFU cache replacement technique, the least frequently used resources can be evicted when the memory is fully used.

2.2.7.3 First-In-First-Out (FIFO)

The first-in-first-out technique (FIFO) sets and maintains a fixed-size queue of tags T. A concrete k-way FIFO cache set S can therefore be modeled as a k-tuple of cache tags, which are ordered from last-in to first-in from left to right [50]. A cache hit does not change the cache set. A cache miss inserts the new tag at position 0, shifting others to the right and evicting the one at the rightmost position. The update function US: $S \times T \rightarrow S$ models the effect of a cache set when accessing a memory block with tag $'T'$[51].

As a cache replacement technique, FIFO has several advantages over other policies of cache replacement [52]. Caches with FIFO replacement technique demonstrate lower energy consumption, especially when compared with another cache using LRU replacement technique [53]. Their simple design makes a FIFO caches inexpensive to implement due to the following reasons: FIFO is widely used as the cache replacement technique in embedded processors such as Tensilica Xtensa LX2 processors, Intel XScale ARM9 and ARM11 processors [52]. Therefore, a fast simulator to decide the cache miss rate of an application on various FIFO caches is in high demand. To meet this demand, smart data structure-based cache simulators are in use [54]. However, the possibility to utilize caching inclusion properties in addition to smart data structures would be of great use in reducing simulation time further. Hence, FIFO cache property has not been significantly reduced to single-pass simulation time [52].

The challenges faced by FIFO in static analysis is that it is difficult to extract may-information [51]. At the same time, may-information is necessary to obtain precise must information. For example, a FIFO cache set has unknown contents. After observing a cache access to a block a, one knows that a must be cached|trivial must information is available. If one cannot classify the access to a as a miss, another access to a different block b may immediately evict a. This is the case if the access to a is a hit on the first-in, i.e., rightmost, position and the access to b is a miss [50]. Thus, without (implicitly or explicitly) classifying some accesses as misses, it is not possible to infer that two or more blocks are cached.

Practical cache replacement policies attempt to follow optimal replacement by predicting the re-reference interval of a cache block. The commonly used LRU replacement technique always predicts a near-immediate re-reference interval on cache hits and misses. Applications exhibiting a distant re-reference interval perform badly under LRU. Such applications usually have a working set larger than the cache or have frequent bursts of references to non-temporal data (called scans). To improve the performance of such workloads, this paper proposes cache replacement using Re-reference Interval Prediction (RRIP). Static RRIP (SRRIP) is scan-resistant while Dynamic RRIP (DRRIP) is both scan-resistant and thrash-resistant. Both RRIP policies require only 2-bits per cache block and easily integrate into existing LRU approximations found in modern processors.

2.3 The Causes of Congestion

The process of controlling and managing congestion is a vital aspect in maintaining network performance to handle packets effectively when it transfers from client to the server. The main causes of congestion have been discussed by many researchers in single and shared communication medium. This is due to that fact that congestion in a network could lead to various system performance problems which include slowing down a network and increase its disturbance. Esmalifalak, Han, and Song [55] explained how network congestion is usually formed, they stated that congestion is established when the load referenced to a certain channel is larger than the supported bandwidth for the same channel.

Current studies have emphasized on the congestion related causes in traditional shared network based on the bandwidth and the network bisection channels [56, 57]. It has been concluded that network congestion can be resulted due to the experience of the bottlenecks found in internal network channels. In addition, network congestion caused can be also related to the lack of using proper replacement schemas in order to shift channels with full bisection bandwidth to randomized channels in the same network [58].

Congestion occurring in networks with sufficient bisection bandwidth can take place at the edge of the network. However, another phase of congestion is a network endpoint which commonly found in wide range networks. The main causes can also be due to the use of incompatible replacement models utilized mostly in large computer systems [59, 60]. In the even that traffic in a network is neglected, the congestion resulted can be randomly effecting the network performance of senders into the same destination. Congestion therefore take place only when the traffic cannot be processed by the over-subscribed destination which results in substantive queues [61].

However, other network schemes associated different network protocols can be affected by the packet congestion, which lead to a packet drop. With this in mind, many studies tried to reduce the congestion in a network by using tightly-controlled buffer allocation policies or replacement schemes in order provide the desired packet flow [62]. This is because ignoring congestion problems without taking the suitable management could affect the whole network performance.

Congestion in a network can be associated with the slow-starting of the dropped packets, which can be reasoned to the unnecessary retransmits due to the failure of the server to manage the resulted bursts. Such even can lead to an unnecessary retransmit timeout [63]. In the event that congestion occurs in a large network, retrieving popular content of packets can without a retransmit timeout, which could also lead to unnecessary transition from the slow-start to the congestion-avoidance phase [64]. Therefore, congestion can also occur when packet drops are experienced in buffering or in moderately-congested networks. The main issues associated with network congestion were identified and summarized into the following:

Duplicate segments

Studies such as [65] revealed that the larger initial window could sometimes lead to an increase in the segment drop from the client start point to the server endpoint. This is due to that packet may not be dropped if the sender had slow-started from an initial server point.

Segments dropped later in the network

Another reason for having congestion is that packets are dropped before it reaches the destination which can be due to the losing of referenced packet's tracks especially in connections that are affected by multiple congested links. In addition, the network may fail to reference the associated packets due to the scarce bandwidth placed usually in the congested link. Therefore, replacement scheme can help solving current issues by introducing better mechanisms for managing packet drops along the network path which lead to congestion collapse.

An increased packet drop rate

The sudden increase in packet drop can be initiated by the interchangeable traffic state which occasionally increase when drop rate it increase further. Some studied mentioned above explained this as the increase substance in drop tail queue that also accompany bursty traffic in times of congestion. Nonetheless, assumed uncorrelated arrivals for unstable network connections could increase the value of the segment drop rate.

2.4 Summary

This chapter discusses previous works on the Information Centric Network, Name Data Networking (NDN) and the basic architecture of Name Data Networking. The cache replacement technique in NDN with respect to the VoD was discussed in detail. A critical review of previous works on cache replacement technique is discussed and presented. However, the chapter shows that the congestion in NDN and VoD is highlighted in the problem streaming. Therefore, there is need to investigate further with the possible technique that could solve the congestion while requesting the online videos, preferably through the replacement approach and through determining the causes of congestion.

CHAPTER THREE
RESEARCH METHODOLOGY

This chapter discusses detailed research methods that would be used to achieve the objectives of the research. It also explains the research process, based on the Design Research Methodology (DRM) adopted for the design, implementation, analysis, and validation of the preferable replacement technique for Named Data Networking (NDN) cache technique to reduce congestion issues during VoD workload. The research process is divided into different stages which are described in detail along with the stages and deliverables. The evaluation of the NDN cache replacement technique in VOD service is conducted by using three key performance metrics of throughput, cache hit ratio, and network load.

3.1 Research Process

The research process is referred to as the number of stages that represent the entire activities in the study. The current study adopts a research process based on the Design Research Methodology (DRM) in order to explain the main activities which give rise to the expected results at the end of the activities. Because it is widely adopted by researchers to conduct effective, efficient, and realistic research [66, 67], DRM can be divided into four stages:

1. Research Clarification (RC).

2. Descriptive Study (DS).

3. Prescriptive Study (PS).

4. Descriptive Study II (DS-II).

The research is divided into four stages in relation to the stages of DRM as illustrated in the following diagram Figure 3.1. The deliverables of each stage are described further.

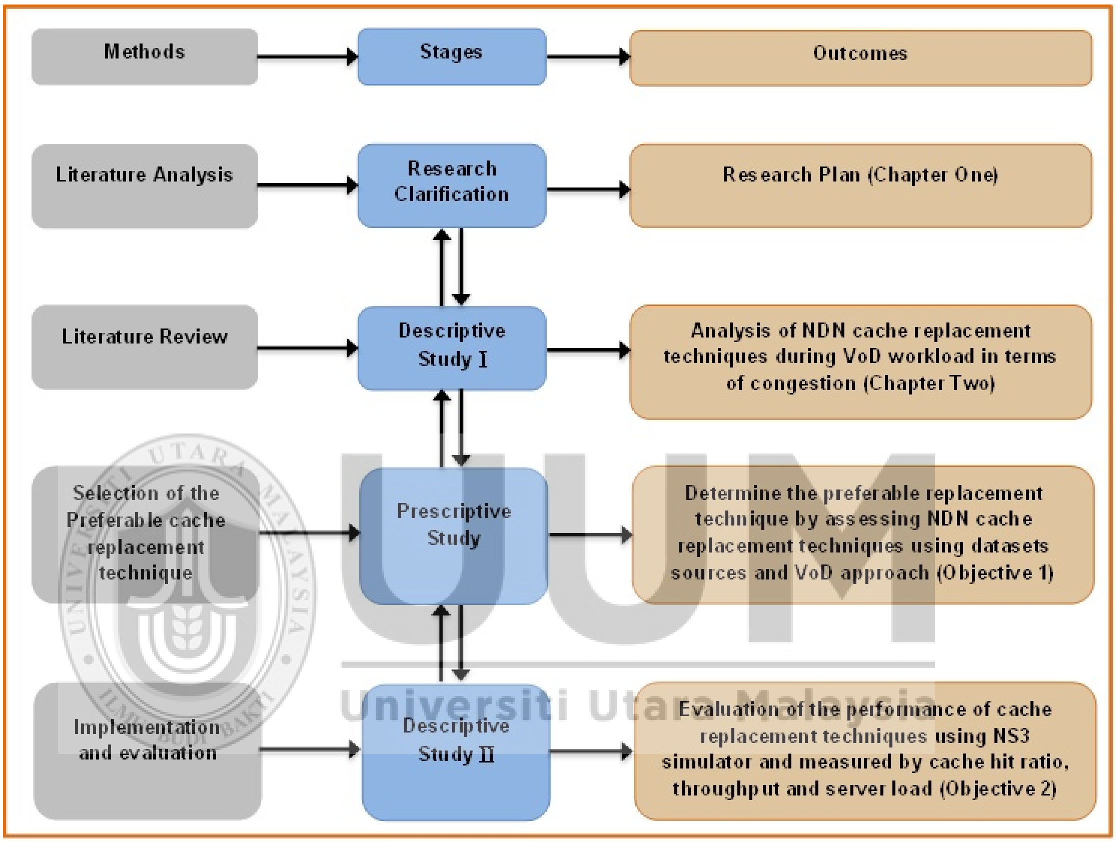


Figure 3.1: Research Process

3.1.1 Stage 1 – Research Clarification (RC)

This is the initial stage of the current research that provides an understanding of the topic while deriving an overall research plan. It provides an extensive analysis of the literatures leading to the formulation of the research problem and the objectives. Whereas, the research steps for this stage are explained in Figure 3.2.

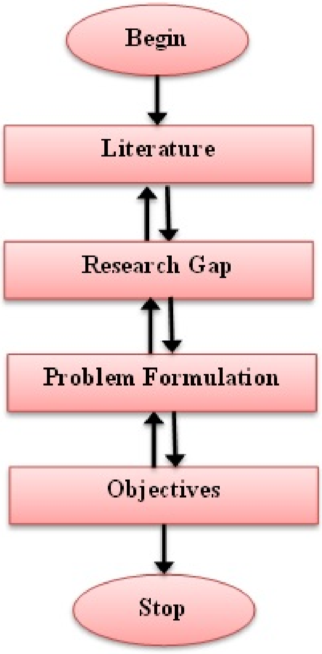


Figure 3.2: Research Steps

The deliverables of this stage are as follows:

- Research problem
- Research questions
- Research objectives
- Research scope and significance

3.1.2 Stage 2 – Descriptive Study (DS)

This stage provides a critical review of the literature in the research area to identify the congestion in the NDN cache techniques during VoD workload. Deliverables of this stage are as follows:

• Critical review of the previous studies on NDN architecture.

• Applications and attributes of NDN.

• Overview of video-on-demand and cache replacement techniques.

• The causes of congestion.

3.1.3 Stage 3 – Prescriptive Study (PS)

This stage aims to determine the preferable cache replacement technique for NDN to improve congestion during VoD workload. To do this, the researcher makes experimantal study of the NDN cache replacement techniques – LRU, LFU, and FIFO – then selects the preferable cache replacement technique to resolve congestion issues during VoD workload. In order to assess the replacement techniques, the study uses datasets sources and VoD approach.

3.1.3.1 PPTV as a Datasets Sources

Data sets are defined as collection of data. To assess the impact or effect of the caching replacement technique in the NDN, this research would obtain the datasets, a router level network topology, and session-level statistics of video views from the operational platform of PPTV. PPTV client instrument software can be used for collecting data from router level topology because this software has traced capabilities [15].

The PPTV is a leading form of online Internet protocol television (IPTV) application in China that allows real-time viewing and downloading of clips streaming. Presently, the PPTV is given a commercial content to the consumers through the server box that relatively has peer-to-peer (P2P) platform delivery [17]. Moreover, researchers have found that PPTV has more than 49.7% of the total number of Internet-based television users in China, thus terming it to be the most used content portals in China.

3.1.3.2 Video on Demand Approach

This approach collects data in the form of a request from the PPTV server logs where each log denotes statistics of a single video request that transfers the log collected whenever the video sessions terminate [16]. Whereas, the following entries represent the categories of operations in the PPTV video clips streaming to be used in assessing the caching replacement techniques in the NDN [15].

Deliverables of this stage are as follows:

• Use datasets sources and VoD approach to assess the caching replacement techniques.

• Implement and assess NDN cache replacement techniques.

• Determine the preferable replacement technique to improve congestion in NDN during VoD workload.

3.1.4 Stage 4 – Descriptive Study II (DS-II)

This stage provides the description of implementation and validation of the cache replacement techniques for NDN. It implements the key performance metrics of cache hit ratio, throughput, and server load to evaluate the performance of the cache replacement techniques of NDN during VoD workload.

The entries of operations of PPTV are Buffering Ratio, Average Bit Rate and Join Time that would be examined while assessing the effect of caching replacement techniques in NDN. Moreover, the following equations below are to understand them briefly.

• Buffering Ratio = Buffering Time / Total View Time. (3.1)

• Average Bit Rate = Total Bytes Downloaded / Total Play Time. (3.2)

• Join Time = Start Time of Play – Request Time. (3.3)

Deliverables of this stage include the following:

• Evaluation of the cache replacement techniques for NDN using the key performance metrics – cache hit ratio, throughput, and server load.

• Validation of the preferable replacement technique by evaluating its performance on resolving the congestion issues of NDN during VoD workload. The Fourth stage of this study uses a simulation approach along with the key performance metrics to evaluate the performance of the cache replacement techniques for NDN. It measures the output of the techniques, analyzes, and compares it with the preferable techniques in order to find out how it performs in improving the congestion issues of NDN during VoD workload.

3.2 Simulation Approach

A simulation is an imitation of the real world for a real system or process which provides the understanding of things around us through creativity and imagination using simulator software. In terms of evaluating real-world performance of a real system, simulations are used for examining the system's behavior under different scenarios in order to find out the preferable of them. Complex in nature, NDN techniques are modeled in a real world which undertakes many assumptions the preferable of which performs the "what if" technique. The best environment for its performance is simulation modeling in which software system models can be designed, analyzed, and evaluated easily. The simulation also can modify and change parts of the system in order to rectify mistakes, if any, or reevaluate the system according to the user's requirements. Simulation modeling is frequently used in NDN networks. Based upon literature survey, the current study uses NS-3 simulators for evaluating the performance of the NDN cache replacement techniques in order to find out the preferable cache replacement technique to improve congestion [68, 69, 70, 71, 72].

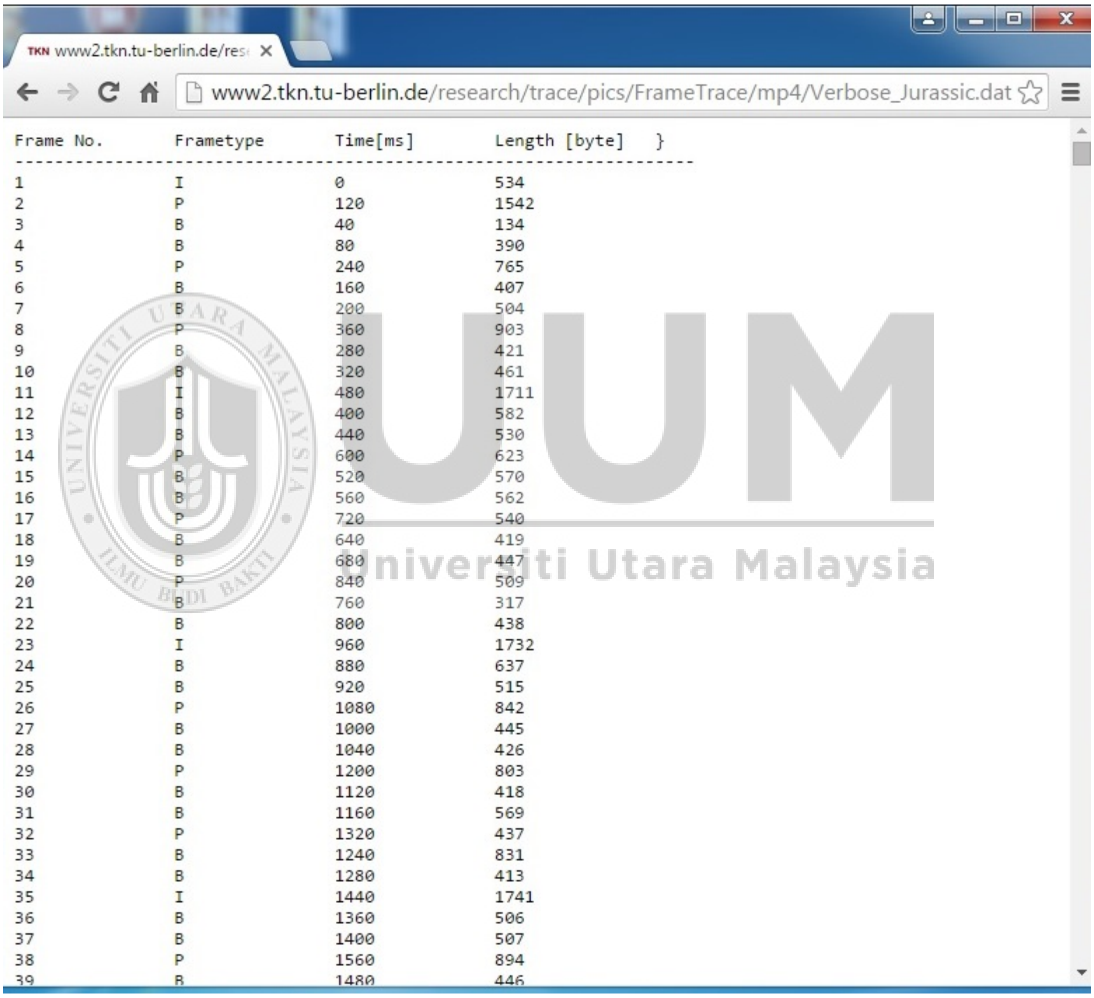
NS-3 is a discrete-event network simulator which implements simulation and models in C++. Functioning as a library, NS-3 is linked statically or dynamically to a C++ main program which further defines the simulation topology and starts the simulator. When it exports an API to Python, Python imports the "ns3" module. Furthermore, NS3 simulation is used for achieving high scalability and extensibility in NDN cache evaluation similar to online VoD services [40, 73]. Also, the packet used of NS3 is an open source and implements NDN protocol stack for network simulation (NS3), that the address location of it is (http://www.nsnam.org/ns-3-22/). Because of the above, the researcher chooses NS3 simulator. Moreover of the high scalability in NDN cache can be achieved through a complete simulation in the case of a very large network with thousands of routers, millions of NDN clients, and infinite number of content-view logs within a reasonable period of time. Whereas, high extensibility in a simulation can be achieved when it supports several existing caching and routing strategies measured by the cache hit ratio, throughput, and server load. This further allows the NDN cache to provide APIs to plug in proposed techniques in a replacement form.

Furthermore, the implementation would be performed through the first and second objectives, whereas the three techniques would be evaluated in term of congestion in first objective and also evaluated in terms of performance metrics in the second objective and comparing them based on cache size (1GB, 10GB, 100GB, and 1TB) [15]. This evaluation is performed by doing the experimental set-up to find the performance of the system. In addition, the output of the simulation techniques will be measured, analysis and making comparison to evaluate them.

The researcher examines the congestion performance of these techniques is based on the standard settings that are usually implemented. In the first objective, the value corresponds to the zero-cache scenario was used as a reference point, whereas VoD through PPTV was used for evaluating three techniques in order to understand the congestion in each of them.

However, the efficiency of the three different replacement techniques needs to be well investigated in order to provide the insights necessary to implement these techniques in certain context. To that end, the researcher implements these three techniques in terms of the certain performance metrics and makes an NS3 analysis of the performance metrics for the above three techniques in terms of cache hit ratio, throughput, and server load results in reasonable outcomes that appears to serve as a potential replacement with the current implementation of the Internet2 topology, where nodes are distributed randomly. However, the results of this implementation enriches the current understanding of replacement techniques in handling different cache sizes (1GB, 10GB, 100GB, and 1TB). After having addressed the different replacement techniques and examined them performance, the performance characteristics along with their expected performance also find to state a cache model for providing a relatively fast running time of across a broad range of embedded applications.

In addition, this empirical implementation sends randomly PPTV packets depended on a trace file of high quality in an MPEG4 stream, that is downloaded from (http://www2.tkn.tu-berlin.de/research/trace/ltvt.html) as shown in Figure 3.3. A valid trace file is a file with 4 columns: 1- the first column represents the frame index. 2- the second column indicates the kind of the frame: I, P or B. 3- the third column refers to the time on which the frame was generated by the encoder. 4- the fourth one refers to the frame size in byte.



Frame No.	Frametype	Time[ms]	Length [byte]
1	I	0	534
2	P	120	1542
3	B	40	134
4	B	80	390
5	P	240	765
6	B	160	407
7	B	200	504
8	P	360	903
9	B	280	421
10	B	320	461
11	I	480	1711
12	B	400	582
13	B	440	530
14	P	600	623
15	B	520	570
16	B	560	562
17	P	720	540
18	B	640	419
19	B	680	447
20	P	840	509
21	B	760	317
22	B	800	438
23	I	960	1732
24	B	880	637
25	B	920	515
26	P	1080	842
27	B	1000	445
28	B	1040	426
29	P	1200	803
30	B	1120	418
31	B	1160	569
32	P	1320	437
33	B	1240	831
34	B	1280	413
35	I	1440	1741
36	B	1360	506
37	B	1400	507
38	P	1560	894
39	B	1480	446

Figure 3.3: Trace File

Moreover, the simulation link of NS3 would ensure a number of routers, clients, videos and many content requests, which provide scalability of the replacement technique [74]. Hence, Table 3.1 that contains the parameters of the simulation, which is used to evaluate the three techniques.

Table 3.1: The Parameters of PPTV Simulation

Description	Value
No. of node	53 nodes
Network area size	562,500 m^2
Simulation time	400 *sec*
Node trannsmeission rang	30 *m*
Average Bit Rate	5 *Mbps*
Join Time	2 *Sec*
Cache size	1*GB*, 10*GB*, 100*GB*, 1*TB*
Packet size	1024 *Bit*

3.3 Performance Metrics

Performance metrics are used for evaluating the performance of the cache replacement techniques for NDN during VoD workload. Based on the literature review [75, 76, 15]. The following metrics are selected for measuring performance :

3.3.1 Cache Hit Ratio

Caches are widely used in computer systems, including CPU level 1 and 2 caches, operating and file system caches, translation look-aside buffers (TLBs), and database (block) buffer caches (Oracle, DB2, Sybase, MySQL, PostgreSQL, InnoDB, etc.). The main function of the cache has been to keep recently used data in a small area, which is faster than the large and slow primary storage area, for future accessing of the data. This results in faster access times for the system. The Cache Hit Ratio refers to the ratio of the number of cache hits to the number of misses, which varies from 60% to greater than 99% depending on the nature of the cache.

The cache hit ratio can be measured by using the following equation:

$$\text{Cache Hit Ratio(CHR)}=\frac{C_H}{C_H+C_M} \quad (3.4)$$

where C_H is number of cache hits and C_M is number of cache misses.

Cache hit ratio is inherently logarithmic, indicating that when it is closer to 100%, its gains are exponentially greater. A cache hit ratio is converted to a relative performance metric by estimating the relative costs of a cache hit and a cache miss.

3.3.2 Throughput

Throughput measures the total amount of information units processed by a system in a given amount of time. Its applications are widespread, ranging from various aspects of computers and network systems in organizations. Related measures of system productivity are: 1) speed with which some specific workload can be completed, 2) response time, and 3) amount of time between a single interactive user request and receipt of the response. In case of storage systems, throughput is defined as the amount of data received and written to the storage medium or read from media and returned to the requesting system. It is typically measured in bytes per second (Bps).

Throughput can be measured by using the following equation:

$$\text{Throughput} = \frac{Size\,of\,information}{Time} \tag{3.5}$$

Another definition of throughput refers to the number of discrete I/O operations responded in a second (IOPS). Throughput can be applied to higher levels of the IT infrastructure where databases and Web servers can be discussed in terms of Transactions Per Second (TPS) and page-views per minute respectively.

3.3.3 Server Load

Server load is defined as the amount of traffic carried by the server. It is expressed in number format. The amount of number of processes standing in chain for seeking processor. Smaller the server loads better the performance of network. But not all processes are similar.

Server load can be measured by using the following equation:

$$SL = \frac{NumCachedReq}{V} \tag{3.6}$$

In the above equation *NumCachedReq* presents the number of requests that are served from some in-network cache. On the other hand, the variable v presents all video requests. In cases of low-priority processes, when a new server request (video request) appears, it can be handled without any delay and responded immediately while the lower priority processes will wait.

3.4 Summary

This chapter has discussed the methodology that would be used for achieving the objectives of this research. The methods contain research process that includes four stages based on the stages of the DRM method. The stages and their deliverables have been discussed in the chapter. The stages included the following: a) an extensive analysis of literatures in order to formulate the research problem, b) critical review of the previous works on NDN architectureto identify the congestion in the NDN cache techniques during VoD workload, c) assessment of the NDN cache replacement techniques using datasets sources and VoD approach in order to determine the preferable cache replacement technique to improve congestion during VoD workload, and d) analysis and evaluation of the performance of the cache replacement techniques which would be conducted using NS3 simulator and measured by three key performance metrics – cache hit ratio, throughput, and server load.

CHAPTER FOUR

RESULTS

4.1 Introduction

ICN concepts are supported in a lot of future Internet architectures such as NDN architecture. One of the main reasons for this behavior is that the Internet traffic consumption is mostly content driven. This means that users mainly worried about the type of content they want, and less about the location of the content that they request. As a result of this, several aspects of the networks nowadays could be simplified. One of the aspects is that complex optimizations that are required by CDNs (Content Delivery Networks) could be eliminated. Another aspect is to provide intrinsic trust in content in order to minimize trust in the sources. The network could be also enabled to scale to greater volumes. This inspiration for NDN goes together with, and is guided in considerable part, by the remarkable increase in Internet video traffic volumes in the last several years. Whereas, NDN caching architecture is presented in Figure 4.1.

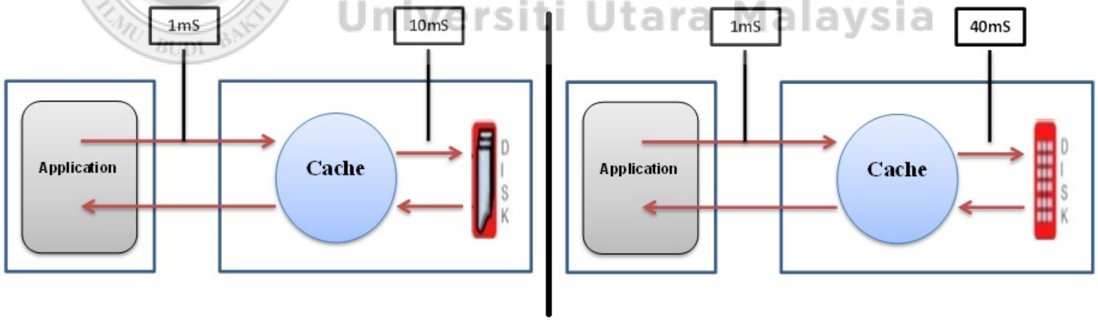


Figure 4.1: NDN Caching Architecture

In this chapter, the researcher evaluated the congestion of well-known types of strategies:

1- Least recently used or LRU.

2- First-in first-out or FIFO.

3- Least frequently used or LFU.

Performance metrics that were considered in this study include cache hit ratio, throughput and server load. In order to ensure the stability of techniques' performance, the researcher considered a combination of 3 content replacement executed over 4 cache size of 1GB, 10GB, 100GB, and 1TB. However, the main concern in this study was to determine the stability of the least congestion technique for a better user experience.

4.2 Dataset and Topology

The basis for establishing the evaluation scenario has a mainly emphasis on the dataset and topology customized in this study for evaluating the selected techniques and to report on the result obtained from such scenario. Datasets, as well as router-level network topology and the statistics of video views were gathered from the PPTV platform. Three types of nodes are usually utilized in order to favor the performance of servers or content origins, and routers.

In this study, the evaluations carried on servers that correspond to the group of the PPTV CDN. It is necessary to note that the mode associated with P2P was not taken into consideration, but only the CDN mode. Actually, 221 servers in total are connected to the network from the same access points as in the real deployment of PPTV. It was expected that these servers are preserving all of the video contents to which the end users were referenced to PPTV clients that are observed in the dataset.

Each of the clients is connected with the access router that has IP with the longest prefix in common. In addition, the researcher considered the same size of the cache. However, the customization of routing tables in each referenced router was injected into the network with the consideration of the shortest paths as recommended by [40, 77]. By doing so, it can be summated that each of the content requests is routed towards the closest origin server. In the event the content is found in an on-path router, then the request is “shortcut”.

Two main dimensions could be researched in this part, the cache size, and the content replacement technique. In this study, the researcher has taken into account 4 cache sizes. So, considering that the total volume of different videos in our dataset is 137 TB, the sizes of caches could store nearly 0.0007, 0.007, 0.07, and 0.7 percentages of the processed contents. As such the value correspond to the zero-cache scenario was applied as a reference point and the following replacement techniques are implemented: FIFO, LRU and LFU.

Moreover, the Internet2 topology was used in this study. Whereas, Internet2 is a research based network, which mainly configured to provide dynamic, innovative and cost-effective hybrid optical and packet network. In contrast to a commercial ISP network, Internet2 issues the information about its network topology and provides complete link traffic data in the form of the round robin database to download. In addition, the round robin database fetches requisite data in a round robin way from multiple data sources and consolidates the data in multiple methods and stores them in Round Robin Archives files.

In addition, the Figure 4.2 is given as the result of the implementation of this study that appears how it works using one server, three towers, 53 cellphones, and medium network area size (562,500 m^2). The Internet2 topology is an exceptional community of the United State (US) and international leaders in research, academia, industry and government. For example, National Lambda Rail (NLR) that is a group of leading US research and technology private companies sector, universities, and dissemination, infrastructure and national networks, a new and unique to promote the advancement of the simultaneous research and applications based on a next-generation network in the field of science, engineering, medicine networks. Also, Internet2 topology has the environment of NDN and cache replacement techniques, therefore it is utilized in this study [78].

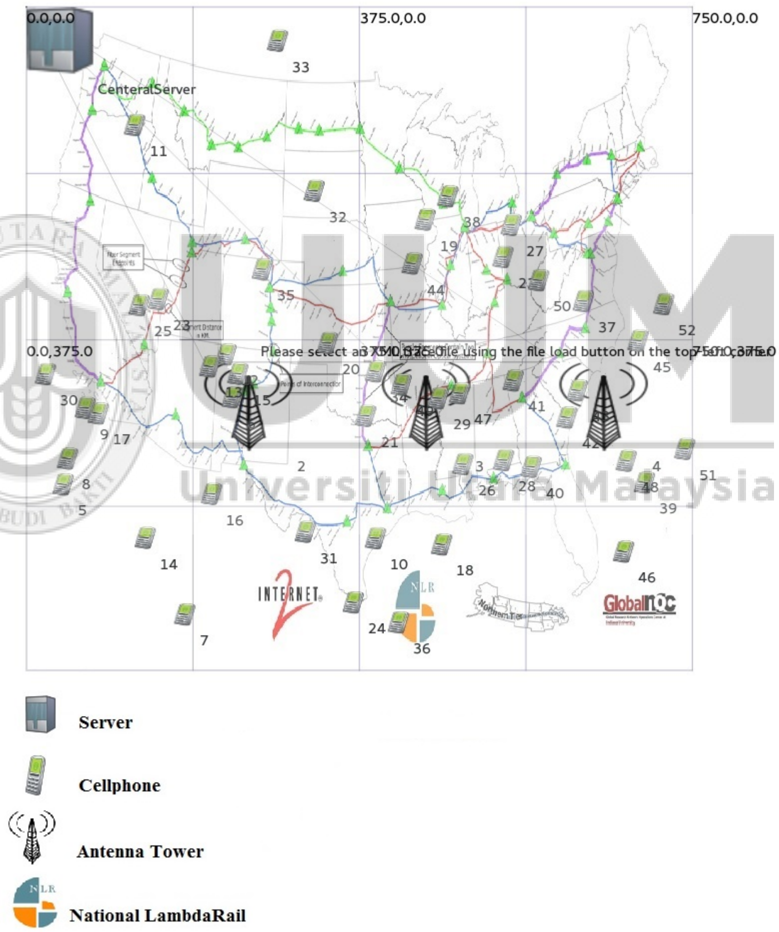


Figure 4.2: Network Topology for PPTV

4.3 Simulation Results

In this study scenario, Video on Demand through PPTV was used for evaluating three techniques, FIFO, LRU and LFU in order to understand the congestion in each of them. The congestion is denoted to requests' content number N_r divided by the number of all interest video requests for all content $\sum_r^R N_r$. Whereas, C_r is refered to a congestion [79].

$$C_r = \frac{N_r}{\sum_r^R N_r} \tag{4.1}$$

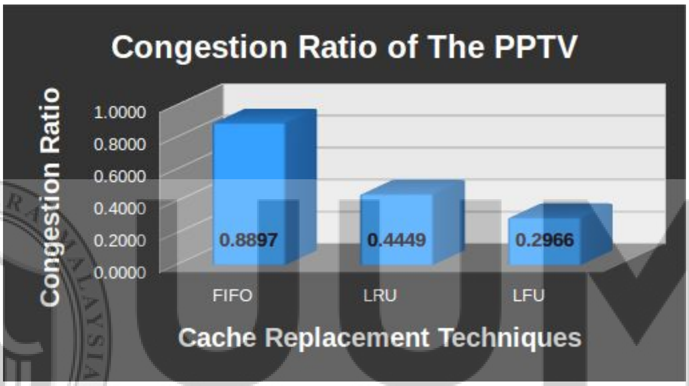


Figure 4.3: Congestion Ratio of The PPTV

According the presented congestion results in Figure 4.3, we can see that the FIFO technique has a value of 0.889706, LRU technique has a value of 0.444853, LFU technique has a value of 0.296569. Hence, LFU has the lowest value of all three presented cache replacement techniques. After all, the selected technique was further tested for other performance metrics discussed in the following section.

4.4 LFU Configuration

The presented scenario in this work where the cache ejection technique was configured based on the information obtained for addressing the frequency value along with the node when request vector is received and compared with the frequency held in the local cache with the requests for other items. On the other hand, the optimal conditions for operating the node regarding to its request vector r was based on caching the items i with the maximum average request rate. If $F_C(r)$ presents the function that returns the C largest values from r, then the cache probability vector for node k, $C_k$as shown in the following equation:

$$C_K^{LFU} = 1_{i \in F_C(r_k)} \tag{4.2}$$

where 1 presents the indicator function.

However, in the event that the average of request rates corresponds to the LFU was static, then the best technique in steady state is to cache items with high average request rates. But, this is not an optimal technique because the temporal variations in requests could initiate ejection of the items that are popular in the long run. As such, modeling the performance of LFU can enable us to understand the estimation of an average case not including the upper or lower performance bounds.

4.5 Performance Evaluation of LFU

4.5.1 Cache Hit Ratio

Average cache hit ratio across all routers could be expressed by the equation (3.4). Whereas, Figure 4.4 shows the cache hit ratio of the techniques selected in this study. The cache hit ratio was tested at 1GB shows a propositional increase in hit ratio starting from LRU which shown to give the highest ratio of 0.008. Both FIFO and LFU shown slightly low ratio of 0.007.

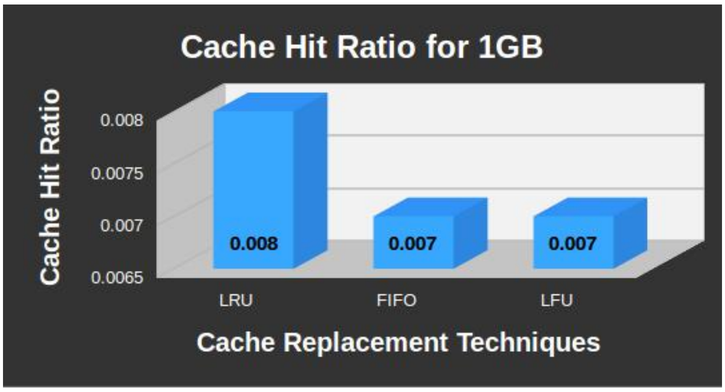


Figure 4.4: Cache Hit Ratio for 1GB

Figure 4.5 shows the result of cache hit ratio for all schemes in 10GB. The result revealed that LFU achieved the highest performance of 0.025 compared to others. The results of other schemes were slightly the same except for LFU which shown to result in the lowest hit ratio.

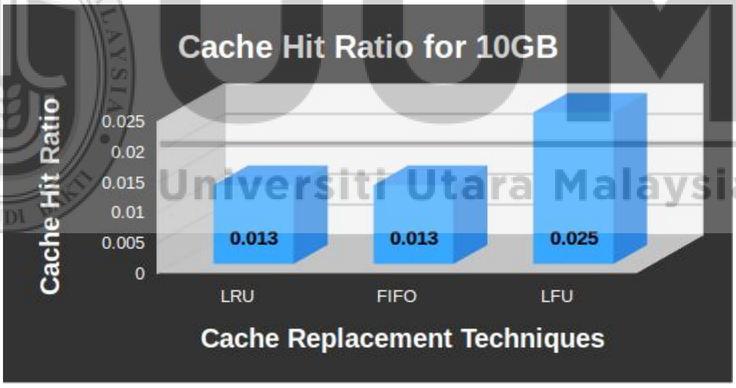


Figure 4.5: Cache Hit Ratio for 10GB

Figure 4.6 shows the result of cache hit ratio for all schemes in 100GB. The result revealed that LFU again performed higher than the other schemes in 0.058. The result of other schemes were approximately same, where the LRU had 0.048 and the FIFO give value of 0.045 that was the lowest hit ratio.

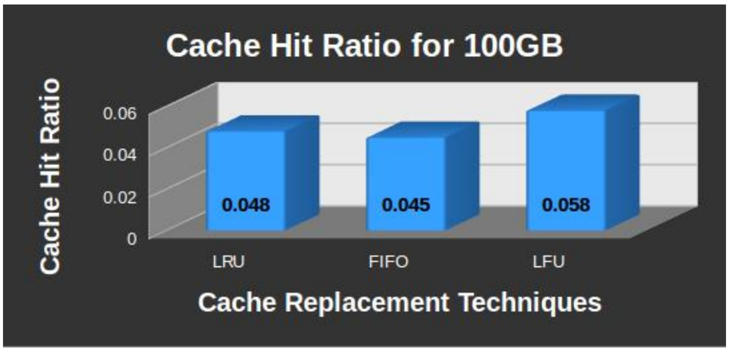


Figure 4.6: Cache Hit Ratio for 100GB

Figure 4.7 presents the result of cache hit ratio for all schemes in 1TB. The result showed that LFU again performed higher than other schemes in 0.111. The result of other schemes were 0.091 hit ratio to LRU but the FIFO was the lowest that had a 0.085 hit ratio.

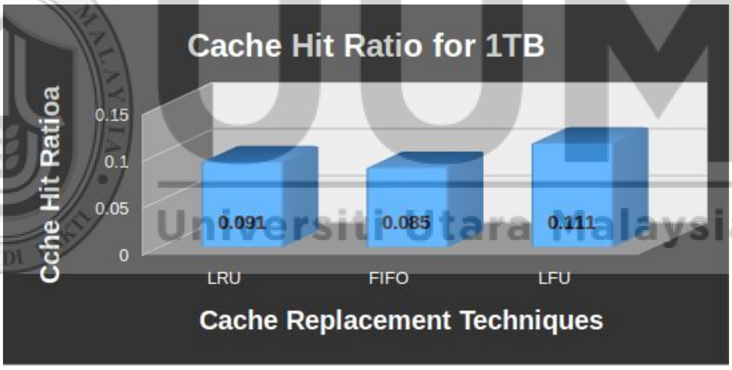


Figure 4.7: Cache Hit Ratio for 1TB

Based on this, it can be concluded that cache hit ratio for the selected schemes were mostly the highest in LFU. The results of the cache hit ratio when the cache size is 1GB can help us to notice that the cache hit ratio of the LFU technique is nearly lowest. Reason for this behavior is that when the cache size is 1GB, the cache hit ratio and the average reduction of video traffic is low across all caching strategies. But, when the cache size is increased to 10GB, the cache hit ratio is significantly improved. LFU to drastically increase.

In addition, Figure 4.8 is aimed to compare the results of each technique based on the four cache sizes. Whereas, the results of cache hit ratio are more than 0.007, 0.013, 0.045, and 0.085 with cache size (1GB, 10GB, 100GB, and 1TB) respectively. Therefore, we can conclude that the significantly increased of the cache hit ratio happens when the cache size is raised. In addition, we can see that the LFU has the best result of cache hit ratio with three cache sizes (10GB, 100GB, and 1TB), because the highest value of cache hit ratio gives the best performance. Therefore, we can conclude that the LFU has the preferable results than the other two techniques.

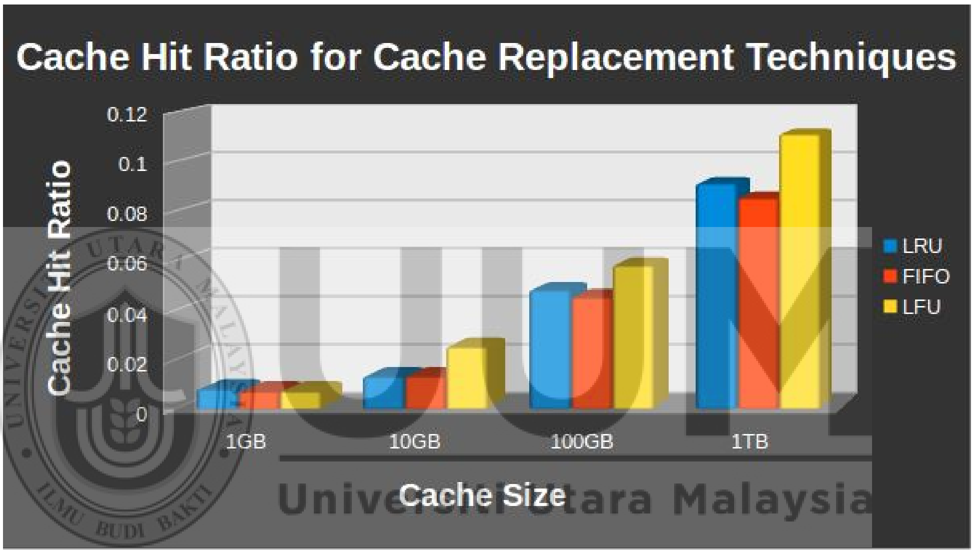


Figure 4.8: Cache Hit Ratio for Cache Replacement Techniques

4.5.2 Analysis the Throughput Performance Metric

The throughput as a performance metric presents the number of bits that are sent in one second. It can be calculated with the following equation:

$$MT = \frac{Bytes*8[bits]}{Time[s]} \tag{4.3}$$

In the above equation Bytes refer to the number of bytes that are sent or received. This value is multiplied by 8 to give the total number of bits, because one byte has 8 bits. The value of Time is expressed in seconds and it refers to the time of the sent or received bytes or bits. So, throughput MT is expressed with the number of transmitting bits per second. Figure 4.9 presents the throughput value for the selected scheme in 1GB. The result showed that LFU provided high throughput value of 125.1, followed by LRU and FIFO respectively.

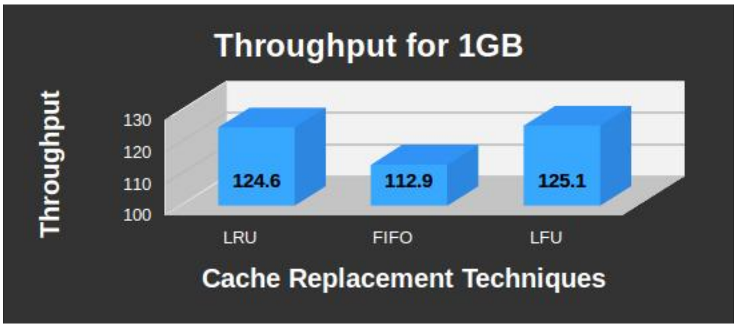


Figure 4.9: Throughput for 1GB

Figure 4.10 presents the throughput value for the selected scheme in 10GB. The result showed that LFU provided the highest throughput value of 126.9. The value of throughput was changed when increasing the size to 10GB. However, LRU provided the lowest throughput value of 121.3.

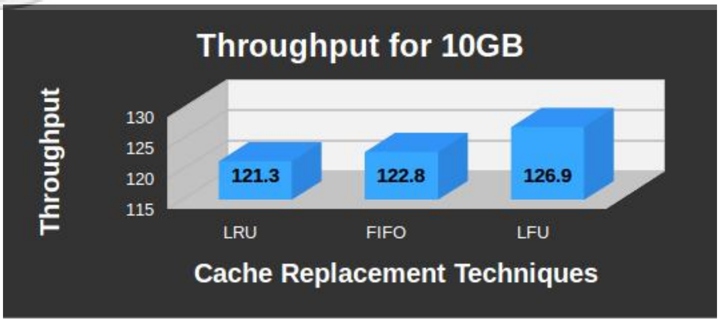


Figure 4.10: Throughput for 10GB

Figure 4.11 presents the throughput value for the selected scheme in 100GB. The result revealed that LFU again has provided the highest throughput value of 120.9. It can be also noticed that the LFU maintained its performance when increasing the size to 100GB. However, LRU provided the lowest throughput value of 115.6.

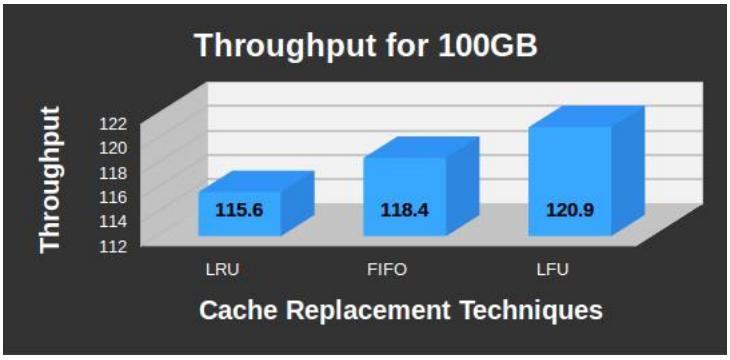


Figure 4.11: Throughput for 100GB

Figure 4.12 presents the throughput value for the selected scheme in 1TB. The result revealed that LFU has provided the highest throughput value of 128.7 followed by FIFO which resulted in 122.2 throughput. In addition, the LRU provided the lowest throughput value of 121.5.

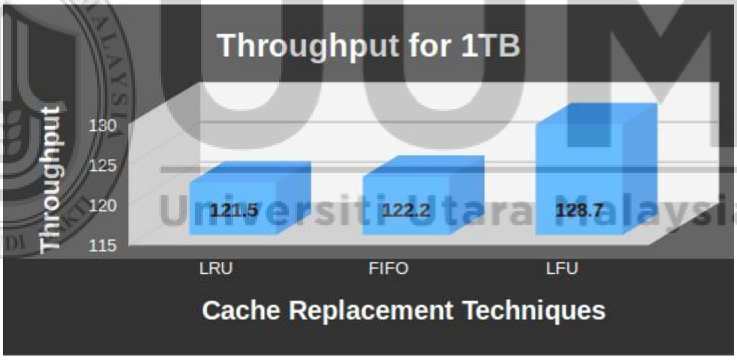


Figure 4.12: Throughput for 1TB

The throughput performance results of the three selected techniques are starting from 1 second to 25 seconds. Figure 4.13 shows a comparison in LRU throughput performance from second 1 to 25. The result shows that the throughput begins with 164.3123 in its early period and ending with 70.6173. The result indicates that LRU throughput performance is constantly reduced when it arrives the second 25. This indicates that LRU is an approach that is affected by the time variance in which the replacing the items source load. However, LRU is driven by the required time to distribute them within the network.

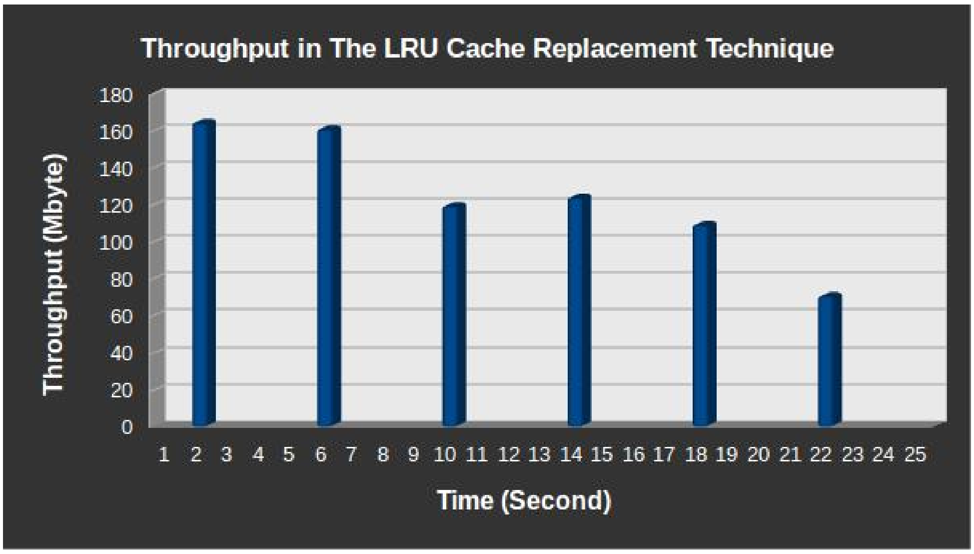


Figure 4.13: Throughput in the LRU Cache Replacement Technique

As for FIFO technique, the results of throughput performance from 1-25 second is shown in Figure 4.14. It is shown that a slight better performance for throughput than LRU. The result indicated that FIFO throughput starts with 169.8762 and ended with 71.7689. This can be reasoned to that FIFO provides an optimal way for replacing resources in a sequential order which donates the received packet with similar path when it sent back. As such, FIFO technique provides better throughput performance than LRU.

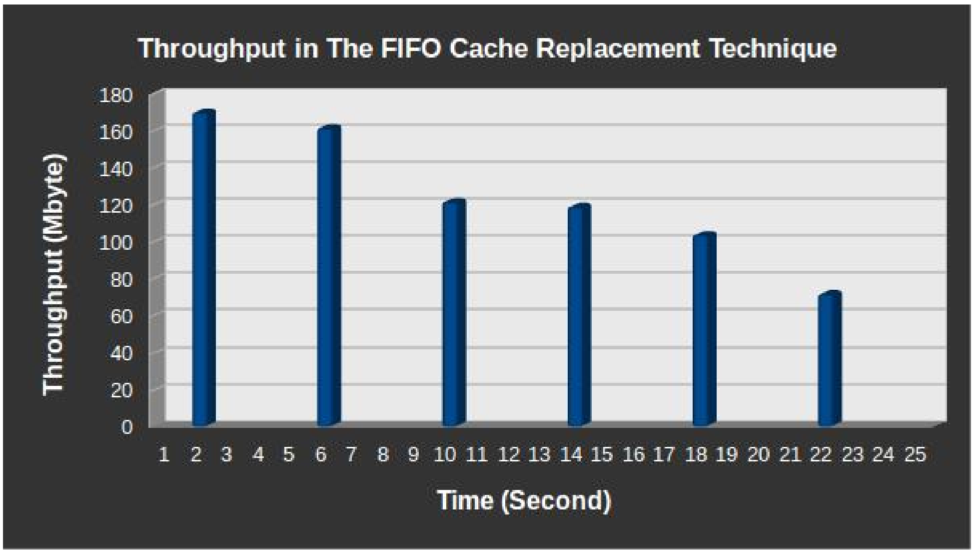


Figure 4.14: Throughput in The FIFO Cache Replacement Technique

Furthermore, Figure 4.15 showed the throughput performance results from 1-25 second of LFU technique. One can note that there is a slight significance difference in LFU throughput performance as compared to the FIFO in which the throughput value was 175.2118 when it firstly started and 70.9931 when it ends. This indicates that LFU provides better throughput performance in its initial stages and ended with lower performance due to its structure. Whereas, it reserves additional space when it progresses. Therefore, it can be concluded that LFU can serve as a good replacement technique.

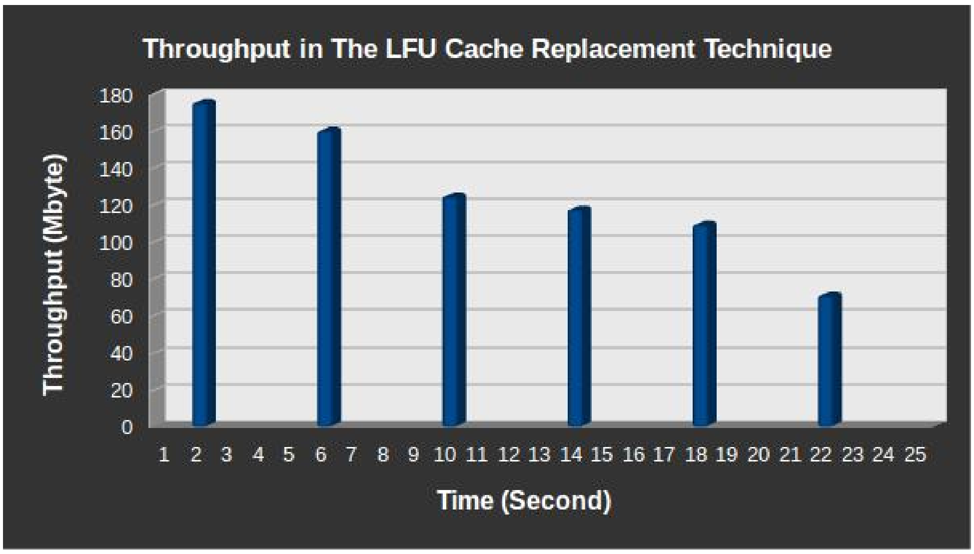


Figure 4.15: Throughput in The LFU Cache Replacement Technique

4.5.3 Analysis of the Performance Metric Server Load

Server load as one of the main performance metrics could be calculated using the equation (3.6). Whereas, Figure 4.16 presents the results of server load expressed in percentage for all three techniques when the cache size is 1GB. It is obvious that the server load is largest when LRU technique is used, and LFU gives the lowest server load value. In this case, LFU technique gives the most favorable results because lower server load gives better overall performances. Anyway, differences between the results of all three techniques in this case are not extremely high, because they are moving between the server load of 4.8% and 5.3%.

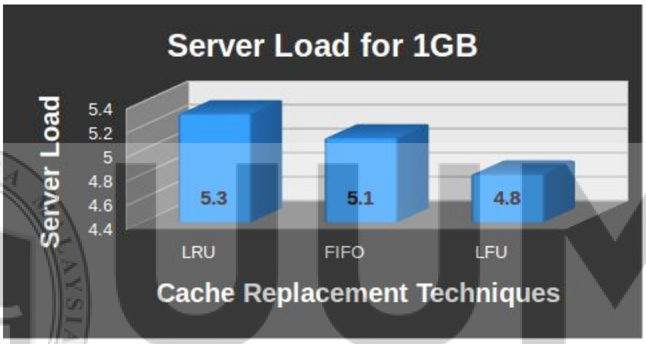


Figure 4.16: Server load for 1GB

Figure 4.17 depicts the results of server load for all three techniques when the cache size is 10GB. Here we can see that server load of the LFU technique is 9.4%, but the server load of LRU and FIFO are almost the same – larger than 14.7%.

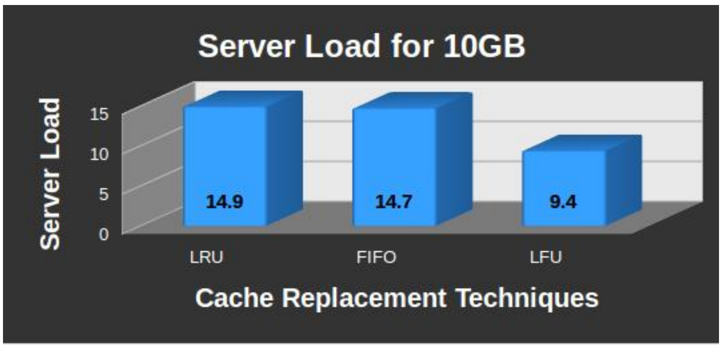


Figure 4.17: Server load for 10GB

Figure 4.18 presents the server load results for all techniques when the cache size is 100GB. From the shown results it is obvious that LFU technique has the greatest server load of around 29.4%. FIFO technique has about 27.9%, and LRU technique has around 28.5% server load.

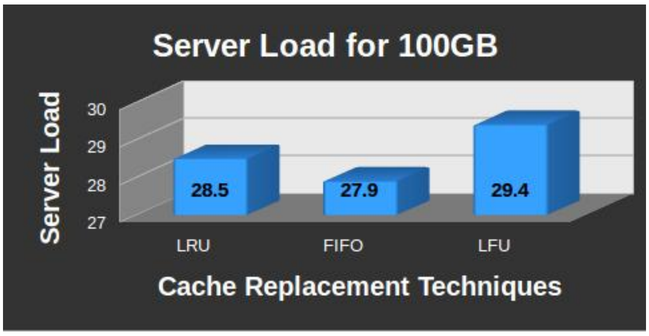


Figure 4.18: Server load for 100GB

Figure 4.19 shows the results of server load of LRU, FIFO, and LFU techniques when the cache size is 1TB. Server load for LFU technique gives the highest results – more than 55%.

LRU technique gives around 46.9% of server load and FIFO around 44%.

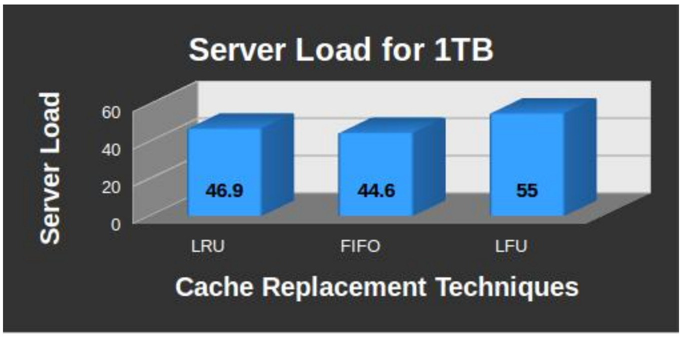


Figure 4.19: Server load for 1TB

From the presented results, generally, we can conclude that the server load is increased when the cache size is increasing for all techniques. We can also confirm that the server load of the LFU technique is the best performance when the cache size is approximately between 1GB to 10GB. Thus, LFU gives the lower server load.

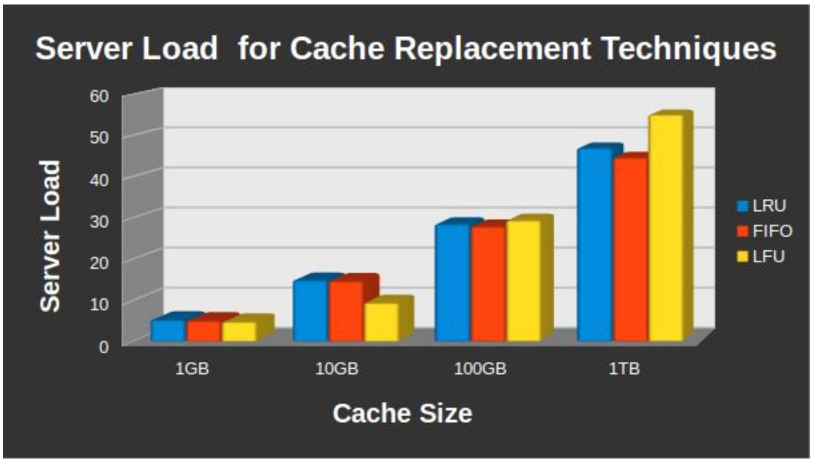


Figure 4.20: Server Load of the Cache Replacement Techniques Expressed in Percentage

Figure 4.20 presents the server load results for the three techniques expressed in percentage for different cache sizes from 1GB to 1TB in order to compare the results of each technique based on the four cache sizes. It is obvious from the diagram that server load is increased as the cache size increases from 1GB to 1TB. It is clearly shown that the server load for all techniques with 1GB is about 5%, but with 1TB is more than 44%, so it can conclude that the best result for server load are with 1GB, because it has a lower server load, which can give better overall performances. Having this in mind, it can be concluded that LFU can offer the favorable results because it has 4.8% that is the lowest percentage of server load with 1GB than the other two techniques (FIFO, and LRU).

4.6 Summary

In this work we have evaluated the widely used cache replacement techniques, LRU, FIFO and LFU in order to understand better the congestion for each of them. We have used Video on Demand through PPTV network to get the necessary results.

From the obtained results we have got the preferable results of congestion for the LFU technique because the congestion was lowest when LFU technique was used in the experiments.

After this part of the work we have examined the performance metrics of cache hit ratio, throughput, and server load for the three techniques. Additionally, we compared the results of this technique in four different cache sizes. In this context, we analyzed and gave particular conclusions for each of the presented diagrams.

Generally, we can conclude that LFU technique, which was the technique with the lowest congestion, gives satisfactory results in the examined performance metrics. It gives the best results for throughput. Compared to other two techniques, we can conclude that LFU technique almost gives the best results in the cache hit ratio. In addition, it has medium performence result server load whereas it gives the favorite result for cache size about 1GB to about 10GB.

CHAPTER FIVE
DISCUSSION

This chapter introduced the discussion of the result obtained and its relation to the current literature from different perspectives. It involves introducing the mechanism of the selected algorithms in a certain scenario related to this study. The assertions by previous studies are also provided to enrich the study result.

5.1 Discussion

A total of three replacement technique names LFU, LRU, and FIFO were evaluated and compared in order to understand better the congestion in NDN network. In this study, the researcher utilized VoD through PPTV network to evaluate the performance of these techniques as explained in the previous chapter. The evaluation result of network performance of these techniques revealed that LFU technique has less congestion that provided optimal results in terms of cache hit ratio.

In addition, LFU was founded to provide the next throughput performance as compared to other protocols, which was reasoned to its capability in locating node redundancy, based on the reference of nodes given for a certain path. However, the server load result showed that LFU shows a significant increment in server load in NDN.

Furthermore, studies like [80] stated that LFU provide the necessary permissions for a network to involve opportunistic caching items divorce, it also provides the permissions for node and also replacement at a constant line-speed. LFU is a specific technique in providing a reliable and stable performance in NDN network where the path is projected to a certain parametric resolution in which direction a number of studies as opposed to caching a similar item from every node [81].

Therefore, one can conclude that LFU as compared to the other protocols is capable of providing permissions for sequencing items in sufficient order for replacement purposes. From this result, it can be summated that LFU is the technique that provides a reliable network performance in NDN in certain aspects. Hence, this study can serve as a reference for future studies to provide in depth understanding the performance of the techniques in NDN network. It also enriches the current body of knowledge about the performance variety of these techniques through PPTV network.

5.1.1 Cache Hit Ratio

The cache hit ratio was evaluated among all the selected techniques through which the cached content was treated as an agent for processing content routers that commonly consists of a distributed index of cached items set under certain topological way. In this study, the researcher found that cache hit ratio was mostly stable in the event of using LFU because it has the best value in three cache size (10GB, 100GB, 1TB). Such result was supported by some previous studies, like [82] who reported the efficiency of LFU in providing a reliable network performance in NDN settings.

However, the focus of their study was to investigate the reliance of certain replacement protocol on caching to provide an adversary basis for executing attacks that are very effective and relatively easy to implement. It can be noticed the domain of the mentioned study was mostly related to examining caching replacement techniques in providing a security feature for the NDN. However, others like [83] asserted that LFU is capable of providing a dependable performance that can offer higher cache hit ratio in NDN network. They acknowledged that LFU during congestion data transfer poses a temporary mapper in which the data is transmitted temporarily for replacement purposes to the source. From this one can conclude that LFU in the current study support the mentioned one in which performance in terms of cache hit ration was driven by the same network parameters.

This case was similar with [84] who declared that network parametric for NDN configuration has constantly impacted the cache hit ratio of the data began distributing in a certain path. This indicates that cache hit ratio for LFU was considered to offer a sufficient variety of performances when comparing it with other techniques, which was reasoned by these studies to its capability in issuing a sequence map for the item of certain resources in congestion with the network configuration. Some previous studies explained that other replacement techniques FIFO and LRU could result in extensive tracking of the positioning of cached objects and if the topological distance to the associated cache is found to be less than the distance towards objects originator the router, then these techniques as a result initiates an identical request number consists of instructions to explicitly request the object from this caching stage. Such perspective can be also associated with the overlay-based caching place in which the stored items are stashed in entities which can be specially created for caching reasons, one popular form is usually web proxy caching. This means that both FIFO and LRU can take more resources for processing the cache request that server requests when its proxies with strategic locations is activated [53, 52].

However, some studies emphasized the LFU performance of the process involved when attaining to a hierarchical and also cooperative caching. It also involves processing containing the items that were originally replicated inside a string of cache, this is because these items are given a sequence order assumed by some sort of cooperation among caches involving different ranges and hierarchical project of priority levels exists in most caching ranges.

On the other hand, cache hit ratio is mostly used to provide a good insight about the network caching performance when the packets are shared in a constant manner based on the dimensionality of the caches. Therefore, the small difference is a significant in cache hit ratio [84]. In addition, studies like [80] stated that LFU provide the necessary permissions for a network to involve opportunistic caching items divorce, it also provides the permissions for node and also replacement at a constant line-speed. Such potential was seen by [81] as a way for balancing caching configuration with the protocol settings by taking into account:

1. The distance factor.

2. Forwarding scheme with shallow flooding.

3. Secondary data source replicator.

Details items are usually cached automatically in each and every router based on the item traverses along while some other items are usually replaced with all the least LFU configuration. Meanwhile, the cache hit ration for LFU scheme has raised doubts and a few authors have already questioned this specific technique in providing a reliable and stable performance in NDN network where the path is projected to a certain parametric resolution in which direction a number of studies as opposed to caching a similar item from every node. Along the process attained when examining the caching that offers a subset of node(s) along the delivery path can do better performance along with server reach rates by simply exploiting the idea of betweenness centrality. Therefore, one can conclude that LFU as compared to the other protocols is capable of providing permissions for sequencing items in sufficient order for replacement purposes.

Moreover, some previous studies, like [85] has descried the applicability of LFU in attaining to NDN's configuration by formulating any potential caching problem into a linear technique which consider the potential advantages of a cache hit instead of simply counting the hit number. This type of caching mechanism pays to in a peer-to-peer sites wherein every single node in the network may become an impartial router and the effective use of an everlasting storage, or the existence of a server can be either not allowed or certainly not profitable to become hosted by way of node in the network.

Therefore, LFU is a certain caching protocol that found to be easily adapted to the information-centric connection scheme and can benefit from a real-time caching mechanism. Where the end users can increase the publication in addition to retrieval associated with notifications about posting items of the cached data. This includes multimedia elements, and other VoD items. On the other hand, LFU confront to the fact that staging the items of the clients in sequential order is helpful in initiating an effective deployment that sometimes can be obliged to help serve these kinds of requests. However, in the event that a new user can reap the benefits of similar likes and dislikes issued by neighboring end users and the cached items for the aim of increasing the performance to retrieve the necessary information/data.

The process of replying involves a certain amount of overhead that may be unavoidable. Note that the cache of node is restricted in which the end users have no idea whether the cached items have reached the destination or not based on the identical period at every single cache. This means that even in case two nodes included clients subscribed for the same filter and have cached in the past, the same items associated with the filter's reference send no certainty that will request only at least one acquire by the same client.

From this, it can be summated that every single item "lives" with a cache is not the same and is dependent upon the nearby workload of the node where LFU can effectively deal with and this is why it offers higher cache hit ratio as compared to the other two techniques.

5.1.2 Throughput

In terms of network throughput performance, the result showed that LFU provided the highest through performance in which the initiated packets are replicated in an effective manner as compared to other two techniques (FIFO, and LRU). The result obtained in this study stands with some previous studies, like [86] who stated that LFU offers a stable throughput performance in NDN network when it is configured to stream items in multiple destinations in-network caching and transcoding services at each NDN node. They considered that the volume for storing and processing associated network resources can be easily scheduled when the LFU is applied to provide solutions for certain optimization problems by balancing the trade-off between the transcoding and bandwidth costs.

The present study is also supported by [87] who found that LFU could offer a redundant array of independent agents that can be used to manage the data compression and list updating hierarchy. Such potential can explain the reason why LFU outperformed other techniques (LRU and FIFO). In addition, the items in these techniques (LRU, and FIFO) are processed, whereas they can still be susceptible to the congestion on account of bandwidth bottleneck difficulty.

Furthermore, throughput performance is affected by the packet switched network congestion like a resource discussing problems which has commonly been posing an important threat to be able to attain to the requirements of NDN networks and the Internet specifically, that contained explosive expansion of bandwidth as well as network targeted visitors load nowadays. In addition, congestion management based cache protocols became one of several major grounds of research nowadays.

A distinctive feature associated with NDN is that it's forwarding planes can discover and recover from network faults alone, enabling every single NDN router to address network breakdowns locally without considering the settings of global routing in such networks. This led the researcher in this study to emphasize that LFU is capable of delivering the highest throughput performance in NDN network.

5.1.3 Server Load

As for the server load, this study found that server load performance in NDN network was favored by using LFU and FIFO techniques. However, the server load result obtained in this study supports some previous studies, like [88] who found that LFU offers a preferable server load based on multitude characteristics of cache replacement techniques because, LFU technique sort the objects based on the number of references. So, it removes the object with the lowest reference count.

However, even if that object is just recently requested, it will be removed while other objects that build up a very high reference count in the past will never be removed. Therefore, the LFU in this study gives better results than LRU and FIFO techniques when the cache sizes are 1GB and 10GB, while it produces the worst results of server load when the cache sizes are 100GB and 1TB. In addition, FIFO techniques give the best server load than LRU and LFU techniques when the cache sizes are 100GB and 1TB. This is because basically the cache size is associated with cache replacement techniques. In the other words, if the size of cache became full of objects, some objects are deleted in order to make space for further objects.

Some other protocols like LRU cache may result in higher sever load when it process the received item's requests at the front end of the actual queue (cache strikes moved upwards from further back in the queue, cache misses inserted at the front end of the actual queue). On the other hand, other protocols like LFU cache consists of individual list that is usually kept with the candidate items to the queue and the amount of requests on their behalf. When the client requests a certain item, it refers to the reference that was firstly assigned to it when it was cached. This means that supplying multiple authentication for data may lead to higher server load as it is also reasoned to the complexity level in the design of these protocols. Therefore, one can conclude that LFU provides a reliable and flexible service for replicating items between the source and the client. From the real time perspective [89] observed that clients of the requested items in NDN network don't frequently transform their funnel when enjoying the live video clip and conferencing in addition takes place in dynamic order. So, entire content interest won't change using every chunk. After all, from the obtained results it can emphasize that the present study result revealed a unstable and reliable data through which the server load of NDN network was reported to be sufficient when using LFU and FIFO.

5.2 Summary

This study discussed and investigated the potential of caching replacement techniques in providing a better network performance when it runs through PPTV in NDN network. The result showed that LFU is considered as the optimal technique for offering reliable cache hit ratio, higher throughput performance and it has a lower server load with 1GB and 10GB. Within distributing fully caching, almost all caches are generally predetermined and mutually associated, but there isn't any central corporation (such as item distribution over multiple node systems). Each cache overflows its buffer opportunistic that dependent on requests from users within its regional community, so there isn't any collaborative work of data to just about every cache. With regards to a cache miss with the local cache, the area associated with data replacement is based on items' association with network path that results in a cache query similar to the one requested by clients. Meanwhile, it should be noted also that from almost all peer are cached before forwarding the request towards the original resource. The began item is processed faster by using LFU than other techniques which can be reasoned to the caching interconnection times as compared to distributed caching.

In addition, other caching techniques have decreased transmission instances than LFU caching. This hypothesized is actually got because a lot of the network site visitors are around less-congested edges with the network. As stated before, the overall performance of replacement techniques consist of the interconnection and transmitting times of caching systems, which usually subjected to high latency. One common caching problem in the present replacement techniques could be the problem connected with caching search engine optimized query outcomes, as well as queries are usually requested by unique users in addition to recompose the outcome every time that could item reach the client with computing assets. From studies of popular replacement caching in NDN network, patterns of usage of clients can be seen to be the main driver of network performance where there appears to be a significant association between the caching technique and requested items.

However, the downfalls in the main aspects of the resilience the selected NDN techniques network was not considered here because of the resource limitation and the significant association between the publish/subscribe net technologies. Similar goals consist of which facts are remained to be offered if the respected origin to be along the initiated resources with enough info on the exterior posted duplicates of these facts undertake by seconds.

In addition, the researcher in this study has mostly focused on the consistency as well as resilience to downfalls by other indicators related to the caching configuration, which may offer superior consistency as well as resilience to downfalls. Therefore, additional works can undertake in order to cover these issues along with the association between the replacement technique and network settings which yet to be discovered.

CHAPTER SIX
CONCLUSION AND FUTURE WORKS

This thesis aims in the empirical evaluation of caching replacement techniques for VoD in NDNs and the effect of these techniques in the congestion performance by NDN via simulation. This chapter provides the conclusion of the research work. It starts with Section 1.1, where the research evaluating three techniques (FIFO, LRU and LFU) in order to understand the congestion in each of them and NS3 analysis the performance metrics for the three techniques. Then, research contributions in Section 1.2 that focuses on the contributions of this research. The limitation of the research is then followed in Section 1.3. Lastly, Section 1.4 contains further studies' suggestions.

6.1 Summary of the Research

This study examined the most common replacement techniques of FIFO, LRU and LFU in terms of giving the least congestion value in Internet2 settings. These techniques were used because of its extensive utilization in desktop applications. The literature showed that using these techniques can significantly stimulate the cache performance to which it also affect the execution cost in terms of gates and power, which is a major point of concern for most computer processors. In order to recommend the best replacement scenario, the researcher examined the congestion performance of these techniques which based on the standard settings that are usually implemented. In addition, NS3 was used to simulate the congestion of these techniques.

In this study, the value corresponds to the zero-cache scenario was used as a reference point, whereas VoD through PPTV was used for evaluating three techniques in order to understand the congestion in each of them. The result showed that LFU technique gives the preferable results for congestion from the presented techniques.

The researcher concluded that the efficiency of the different replacement techniques needs to be well investigated in order to provide the insights necessary to implement these techniques in certain context. To that end, an NS3 analysis of the performance metrics for the above three techniques in terms of cache hit ration, throughput, and server load results in reasonable outcomes that appears to serve as a potential replacement with the current implementation of the Internet2 topology where nodes are distributed randomly. This result enriches the current understanding of replacement techniques in handling different cache sizes.

After having addressed the different replacement techniques and examined them performance, the performance characteristics along with their expected performance were also found to stimulate a cache model for providing a relatively fast running time of across a broad range of embedded applications.

Finally, the result shows that the finding of bottlenecks in Internal network channels, and the lacking of using proper replacement schema are the main reasons of congestion. Also the research summated that the performance metrics of cache hit ratio, throughput, and server load for the LFU with the lowest congestion to be sufficient.

6.2 Research Contributions

In general, This study has a practical contribution to request for more Internet videos once the depot that is less data congestion. In addition, it gives positive affects to both providers and users of Internet videos. In other meanings, there would be more revenue for the Internet service providers and indirectly the host of online video once the long and reducing the frequent buffering of the request of VoD workload, if not completely stopped. Moreover, the study helps academically merging it with the previous works on NDN, online video and Internet of Thing.

6.3 Research Limitation

This work contains several limitations through management the ICN whereas ICN is the future Internet idea. The NDN researches appear as one of the many ICN research and all of them are still under research, while this research was conducted to determine the performance of least congestion replacement technique in Internt 2 context. The result showed that modifying data caches based on the replacement techniques can improve the efficiency of the system in a more significant way, which can be reasoned to the average access time resulted from node distribution.

Some limitations can be noticed in this study, these include applying certain levels of caches to which each level has its own characteristics. As such, the associativity and the length of execution are the main parameters which change and necessarily for having a different constraints to the two options. Some studies happen to be performed based on the association to desktop systems which are not begun optimized for embedded processors' characteristics.

6.4 Future Works

The future studies can consider additional cache sizes along with the invalidation of the coherence protocol to which it may affect the object of node replacement. In addition, future work can consider the probability of hit as it is equivalent to the information of stack data that are modified or exclusive kept in the cache to share the core. Of course, this is a potential error if the data will be adjusted by certain processor, hence, more work can be done to examine the role of such replacement techniques on the performance of multiprocessing systems.

REFERENCES

[1] 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," *NDN-0001, Xerox Palo Alto Research Center-PARC*, 2010.

[2] J. Burke, "Video streaming over named data networking," *E-LETTER*, vol. 8, no. 4, pp. 6–9, 2013.

[3] C. Bian, Z. Zhu, A. Afanasyev, E. Uzun, and L. Zhang, "Deploying key management on ndn testbed," *UCLA, Peking University and PARC, Tech. Rep*, 2013.

[4] 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*, pp. 1–12, ACM, 2009.

[5] 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.

[6] Z. Zhu and A. Afanasyev, "Let's chronosync: Decentralized dataset state synchronization in named data networking.," in *ICNP*, pp. 1–10, 2013.

[7] C. Huang, J. Li, and K. W. Ross, "Can internet video-on-demand be profitable?," *ACM SIGCOMM Computer Communication Review*, vol. 37, no. 4, pp. 133–144, 2007.

[8] L. Gomes, "Will all of us get our 15 minutes on a youtube video?," *Wall Street Journal*, vol. 30, p. B1, 2006.

[9] A. Ali-Eldin, M. Kihl, J. Tordsson, and E. Elmroth, "Analysis and characterization of a video-on-demand service workload," in *Proceedings of the 6th ACM Multimedia Systems Conference*, pp. 189–200, ACM, 2015.

[10] 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," *Communications Surveys & Tutorials, IEEE*, vol. 16, no. 2, pp. 1024–1049, 2014.

[11] 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*, p. 1, ACM, 2011.

[12] A. Smith, "The internets role in campaign 2008," *Pew Internet & American Life Project*, vol. 15, 2009.

[13] T.-Y. Huang, R. Johari, N. McKeown, M. Trunnell, and M. Watson, "A buffer-based approach to rate adaptation: Evidence from a large video streaming service," in *Proceedings of the 2014 ACM conference on SIGCOMM*, pp. 187–198, ACM, 2014.

[14] J. Liu, A. Panda, A. Singla, B. Godfrey, M. Schapira, and S. Shenker, "Ensuring connectivity via data plane mechanisms.," in *NSDI*, pp. 113–126, 2013.

[15] Y. Sun, S. K. Fayaz, Y. Guo, V. Sekar, Y. Jin, M. A. Kaafar, and S. Uhlig, "Trace-driven analysis of icn caching algorithms on video-on-demand workloads," in *Proceedings of the 10th ACM International on Conference on emerging Networking Experiments and Technologies*, pp. 363–376, ACM, 2014.

[16] Z. Ming, M. Xu, and D. Wang, "Age-based cooperative caching in information-centric networks," in *Computer Communications Workshops (INFOCOM WKSHPS), 2012 IEEE Conference on*, pp. 268–273, IEEE, 2012.

[17] S. Salsano, A. Detti, M. Cancellieri, M. Pomposini, and N. Blefari-Melazzi, "Transport-layer issues in information centric networks," in *Proceedings of the second edition of the ICN workshop on Information-centric networking*, pp. 19–24, ACM, 2012.

[18] A. Balachandran, V. Sekar, A. Akella, and S. Seshan, "Analyzing the potential benefits of cdn augmentation strategies for internet video workloads," in *Proceedings of the 2013 conference on Internet measurement conference*, pp. 43–56, ACM, 2013.

[19] L. Saino, I. Psaras, and G. Pavlou, "Icarus: a caching simulator for information centric networking (icn)," in *Proceedings of the 7th International ICST Conference on Simulation Tools and Techniques*, pp. 66–75, ICST (Institute for Computer Sciences, Social-Informatics and Telecommunications Engineering), 2014.

[20] G. Bianchi, A. Detti, A. Caponi, and N. Blefari, "Check before storing: what is the performance price of content integrity verification in lru caching?," *ACM SIGCOMM Computer Communication Review*, vol. 43, no. 3, pp. 59–67, 2013.

[21] H. Jun, "Icnrg j. hong internet draft etri intended status: Informational w. chun expires: January 2015 hufs,"

[22] D. A. Farber and R. D. Lachman, "Data processing system using substantially unique identifiers to identify data items, whereby identical data items have the same identifiers," Nov. 2 1999. US Patent 5,978,791.

[23] D. O. Coileain and D. Omahony, "Accounting and accountability in content distribution architectures: A survey," *ACM Computing Surveys (CSUR)*, vol. 47, no. 4, p. 59, 2015.

[24] J. Ren, L. Li, H. Chen, S. Wang, S. Xu, G. Sun, J. Wang, and S. Liu, "On the deployment of information-centric network: Programmability and virtualization," in *Computing, Networking and Communications (ICNC), 2015 International Conference on*, pp. 690–694, IEEE, 2015.

[25] R.-I. Ciobanu, C. Dobre, and F. Xhafa, "Data modeling for socially based routing in opportunistic networks," in *Modeling and Processing for Next-Generation Big-Data Technologies*, pp. 29–55, Springer, 2015.

[26] E. AbdAllah, H. S. Hassanein, and M. Zulkernine, "A survey of security attacks in information-centric networking," *IEEE*, pp. 1441 – 1454, 2015.

[27] P. Gasti, G. Tsudik, E. Uzun, and L. Zhang, "Dos and ddos in named data networking," in *Computer Communications and Networks (ICCCN), 2013 22nd International Conference on*, pp. 1–7, IEEE, 2013.

[28] N. Fotiou, Y. Thomas, V. A. Siris, and G. C. Polyzos, "Security requirements and solutions for integrated satellite-terrestrial information-centric networks," in *Advanced Satellite Multimedia Systems Conference and the 13th Signal Processing for Space Communications Workshop (ASMS/SPSC), 2014 7th*, pp. 106–113, IEEE, 2014.

[29] G. Paul, F. Hutchison, and J. Irvine, "Security of the maidsafe vault network," in *Wireless World Research Forum Meeting 32 (WWRF32)*, 2014.

[30] 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*, pp. 27–36, ACM, 2014.

[31] 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*, pp. 15–26, ACM, 2012.

[32] R. Margolies, A. Sridharan, V. Aggarwal, R. Jana, N. Shankaranarayanan, V. A. Vaishampayan, and G. Zussman, "Exploiting mobility in proportional fair cellular scheduling: Measurements and algorithms," in *INFOCOM, 2014 Proceedings IEEE*, pp. 1339–1347, IEEE, 2014.

[33] J. Erman, A. Gerber, K. Ramadrishnan, S. Sen, and O. Spatscheck, "Over the top video: the gorilla in cellular networks," in *Proceedings of the 2011 ACM SIGCOMM conference on Internet measurement conference*, pp. 127–136, ACM, 2011.

[34] A. Afanasyev, P. Mahadevan, I. Moiseenko, E. Uzun, and L. Zhang, "Interest flooding attack and countermeasures in named data networking," in *IFIP Networking Conference, 2013*, pp. 1–9, IEEE, 2013.

[35] Q. Xu, S. Mehrotra, Z. Mao, and J. Li, "Proteus: network performance forecast for real-time, interactive mobile applications," in *Proceeding of the 11th annual international conference on Mobile systems, applications, and services*, pp. 347–360, ACM, 2013.

[36] R. K. Mok, X. Luo, E. W. Chan, and R. K. Chang, "Qdash: a qoe-aware dash system," in *Proceedings of the 3rd Multimedia Systems Conference*, pp. 11–22, ACM, 2012.

[37] Z. Zhu, C. Bian, A. Afanasyev, V. Jacobson, and L. Zhang, "Chronos: Serverless multi-user chat over ndn," tech. rep., NDN, Technical Report NDN-0008, 2012.

[38] 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.

[39] J. Jiang, V. Sekar, and H. Zhang, "Improving fairness, efficiency, and stability in http-based adaptive video streaming with festive," in *Proceedings of the 8th international conference on Emerging networking experiments and technologies*, pp. 97–108, ACM, 2012.

[40] S. K. Fayazbakhsh, Y. Lin, A. Tootoonchian, A. Ghodsi, T. Koponen, B. Maggs, K. Ng, V. Sekar, and S. Shenker, "Less pain, most of the gain: Incrementally deployable icn," in *ACM SIGCOMM Computer Communication Review*, vol. 43, pp. 147–158, ACM, 2013.

[41] J. R. P. S. A. Ghodsi, T. Koponen and S. Shenker, "Naming in content-oriented architectures," in *Proceedings of the ACM SIGCOMM workshop on Information-centric networking*, pp. 1–6, ACM, 2011.

[42] K. D. Babu, M. Somu, and N. Rengarajan, "A replacement policy for buffer management in iptv services," 2013.

[43] Cisco, "Cisco visual networking index: Forecast and methodology," 2014.

[44] Citrix, "Mobile analytics report," 2014.

[45] K. Arora and D. Rao, "Web cache page replacement by using lru and lfu algorithms with hit ratio: A case unification.," *International Journal of Computer Science & Information Technologies*, vol. 5, no. 3, 2014.

[46] K. Shah, A. Mitra, and D. Matani, "An 0 (1) algorithm for implementing the lfu cache eviction scheme," tech. rep., Technical report, 2010." http://dhruvbird.com/lfu. pdf, 2010.

[47] A. Patil, M. Prakash, and A. Nimkar, "First-in not referenced first-out page replacement algorithm," in *Proceedings of the International Conference & Workshop on Emerging Trends in Technology*, pp. 443–446, ACM, 2011.

[48] W. Ali, S. M. Shamsuddin, and A. S. Ismail, "A survey of web caching and prefetching," *International Journal of Advances in Soft Computing and its Application*, vol. 3, no. 1, pp. 18–44, 2011.

[49] M. K. Madi, *Replica Creation Algorithm for Data Grids*. PhD thesis, Universiti Utara Malaysia, 2012.

[50] B. Jeannet, T. Jeron, and T. Le, "Abstract interpretation of fifo channels," 2005.

[51] D. Grund and J. Reineke, "Precise and efficient fifo-replacement analysis based on static phase detection," in *Real-Time Systems (ECRTS), 2010 22nd Euromicro Conference on*, pp. 155–164, IEEE, 2010.

[52] M. S. Haque, J. Peddersen, and S. Parameswaran, "Ciparsim: Cache intersection property assisted rapid single-pass fifo cache simulation technique," in *Proceedings of the International Conference on Computer-Aided Design*, pp. 126–133, IEEE Press, 2011.

[53] N. Duong, R. Cammarota, D. Zhao, T. Kim, and A. Veidenbaum, "Score: A score-based memory cache replacement policy," in *JWAC 2010-1st JILP Worshop on Computer Architecture Competitions: cache replacement Championship*, 2010.

[54] M. S. Haque, J. Peddersen, A. Janapsatya, and S. Parameswaran, "Dew: a fast level 1 cache simulation approach for embedded processors with fifo replacement policy," in *Proceedings of the Conference on Design, Automation and Test in Europe*, pp. 496–501, European Design and Automation Association, 2010.

[55] M. Esmalifalak, Z. Han, and L. Song, "Effect of stealthy bad data injection on network congestion in market based power system," in *Wireless Communications and Networking Conference (WCNC), 2012 IEEE*, pp. 2468–2472, IEEE, 2012.

[56] N. Jiang, D. U. Becker, G. Michelogiannakis, and W. J. Dally, "Network congestion avoidance through speculative reservation," in *High Performance Computer Architecture (HPCA), 2012 IEEE 18th International Symposium on*, pp. 1–12, IEEE, 2012.

[57] D. Kliazovich, P. Bouvry, and S. U. Khan, "Dens: data center energy-efficient network-aware scheduling," *Cluster computing*, vol. 16, no. 1, pp. 65–75, 2013.

[58] H. Khadilkar and H. Balakrishnan, "Network congestion control of airport surface operations," *Journal of Guidance, Control, and Dynamics*, vol. 37, no. 3, pp. 933–940, 2014.

[59] A. E. Cohen, J.-H. Lin, and K. K. Parhi, "Variable data rate (vdr) network congestion control (ncc) applied to voice/audio communication," *Computer Networks*, vol. 56, no. 4, pp. 1343–1356, 2012.

[60] R. Cominetti and C. Guzman, "Network congestion control with markovian multipath routing," *Mathematical Programming*, vol. 147, no. 1-2, pp. 231–251, 2014.

[61] M. Rossi, G. Vigano, and D. Moneta, "Hosting capacity of distribution networks: Evaluation of the network congestion risk due to distributed generation," in *Clean Electrical Power (ICCEP), 2015 International Conference on*, pp. 716–722, IEEE, 2015.

[62] C. P. Sahu, P. S. Yadav, S. Ahuja, R. Prasad, and A. Garg, "Optimistic congestion control to improve the performance of mobile ad hoc network," in *Advance Computing Conference (IACC), 2013 IEEE 3rd International*, pp. 394–398, IEEE, 2013.

[63] M. H. Cheung, R. Southwell, and J. Huang, "Congestion-aware network selection and data offloading," in *Information Sciences and Systems (CISS), 2014 48th Annual Conference on*, pp. 1–6, IEEE, 2014.

[64] V. Joseph and G. De Veciana, "Stochastic networks with multipath flow control: impact of resource pools on flow-level performance and network congestion," *ACM SIGMETRICS Performance Evaluation Review*, vol. 39, no. 1, pp. 61–72, 2011.

[65] D. Kim, S. Yoo, and S. Lee, “A network congestion-aware memory subsystem for manycore,” *ACM Transactions on Embedded Computing Systems (TECS)*, vol. 12, no. 4, p. 110, 2013.

[66] L. T. Blessing and A. Chakrabarti, *DRM, a design research methodology*. Springer Science & Business Media, 2009.

[67] A. M. M. Habbal and S. Hassan, “A model for congestion control of transmission control protocol in mobile wireless ad hoc networks,” *Journal of Computer Science*, vol. 9, no. 3, p. 335, 2013.

[68] G. Grassi, D. Pesavento, G. Pau, L. Zhang, and S. Fdida, “Navigo: Interest forwarding by geolocations in vehicular named data networking,” *arXiv preprint arXiv:1503.01713*, 2015.

[69] 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, 2015.

[70] H. Park, H. Jang, and T. Kwon, “Popularity-based congestion control in named data networking,” in *Ubiquitous and Future Networks (ICUFN), 2014 Sixth International Conf on*, pp. 166–171, IEEE, 2014.

[71] D. Posch, B. Rainer, and H. Hellwagner, “Saf: Stochastic adaptive forwarding in named data networking,” *arXiv preprint arXiv:1505.05259*, 2015.

[72] K. Wang, J. Chen, H. Zhou, Y. Qin, and H. Zhang, “Modeling denial-of-service against pending interest table in named data networking,” *International Journal of Communication Systems*, vol. 27, no. 12, pp. 4355–4368, 2014.

[73] B. Han, X. Wang, N. Choi, T. Kwon, and Y. Choi, “Amvs-ndn: Adaptive mobile video streaming and sharing in wireless named data networking,” in *Computer Communications Workshops (INFOCOM WKSHPS), 2013 IEEE Conference on*, pp. 375–380, IEEE, 2013.

[74] L. Saino, I. Psaras, and G. Pavlou, “Hash-routing schemes for information centric networking,” in *Proceedings of the 3rd ACM SIGCOMM workshop on Information-centric networking*, pp. 27–32, ACM, 2013.

[75] F. Dobrian, A. Awan, D. Joseph, A. Ganjam, J. Zhan, V. Sekar, I. Stoica, and H. Zhang, “Understanding the impact of video quality on user engagement,” *Communications of the ACM*, vol. 56, no. 3, pp. 91–99, 2013.

[76] N. Lv, D. Zhang, *et al.*, “On performance of cache policies in named data networking,” in *2013 International Conference on Advanced Computer Science and Electronics Information (ICACSEI 2013)*, Atlantis Press, 2013.

[77] G. Tyson, S. Kaune, S. Miles, Y. El-khatib, A. Mauthe, and A. Taweel, “A trace-driven analysis of caching in content-centric networks,” in *Computer Communications and Networks (ICCCN), 2012 21st International Conference on*, pp. 1–7, IEEE, 2012.

[78] X. Zheng, M. Veeraraghavan, N. S. Rao, Q. Wu, and M. Zhu, "Cheetah: Circuit-switched high-speed end-to-end transport architecture testbed," 2005.

[79] M. Badov, A. Seetharam, J. Kurose, V. Firoiu, and S. Nanda, "Congestion-aware caching and search in information-centric networks," in *Proceedings of the 1st international conference on Information-centric networking*, pp. 37–46, ACM, 2014.

[80] T. Lauinger, N. Laoutaris, P. Rodriguez, T. Strufe, E. Biersack, and E. Kirda, "Privacy implications of ubiquitous caching in named data networking architectures," tech. rep., Technical report, TR-iSecLab-0812-001, iSecLab, 2012.

[81] S. Wang, J. Bi, J. Wu, Z. Li, W. Zhang, and X. Yang, "Could in-network caching benefit information-centric networking?," in *Proceedings of the 7th Asian Internet Engineering Conference*, pp. 112–115, ACM, 2011.

[82] M. Conti, P. Gasti, and M. Teoli, "A lightweight mechanism for detection of cache pollution attacks in named data networking," *Computer Networks*, vol. 57, no. 16, pp. 3178–3191, 2013.

[83] Y. Wang, Z. Li, G. Tyson, S. Uhlig, and G. Xie, "Optimal cache allocation for content-centric networking," in *Network Protocols (ICNP), 2013 21st IEEE International Conference on*, pp. 1–10, IEEE, 2013.

[84] J. Ran, N. Lv, D. Zhang, Y. Ma, and Z. Xie, "On performance of cache policies in named data networking," in *International Conference on Advanced Computer Science and Electronics Information*, pp. 668–671, 2013.

[85] S. Wang, J. Bi, and J. Wu, "On performance of cache policy in information-centric networking," in *Computer Communications and Networks (ICCCN), 2012 21st International Conference on*, pp. 1–7, IEEE, 2012.

[86] Y. Jin and Y. Wen, "Paint: Partial in-network transcoding for adaptive streaming in information centric network," in *Quality of Service (IWQoS), 2014 IEEE 22nd International Symposium of*, pp. 208–217, IEEE, 2014.

[87] N. Megiddo and D. S. Modha, "Outperforming lru with an adaptive replacement cache algorithm," *Computer*, vol. 37, no. 4, pp. 58–65, 2004.

[88] E.-S. M. El-Alfy and A. Orwani, "The need for a unified framework for evaluating web cache replacement strategies," in *GCC Conference & Exhibition, 2009 5th IEEE*, pp. 1–5, IEEE, 2009.

[89] C. Yao, L. Fan, Z. Yan, and Y. Xiang, "Long-term interest for realtime applications in the named data network," *Proceedings of the AsiaFI*, 2012.