논문 19-44-10-12 The Journal of Korean Institute of Communications and Information Sciences '19-10 Vol.44 No.10
https://doi.org/10.7840/kics.2019.44.10.1860

# Cache Performance Investigation in NDN for Large Scale Scientific Data

Inayat Ali◆, Huhnkuk Lim°

## ABSTRACT

Named-Data Networking (NDN) is the most active instance of Information-centric Networking (ICN) having a very active research community. NDN is racing for the future Internet architecture by removing dependence on IP addresses, location, and host-to-host communication model. The communication paradigm in NDN revolves around the content/data by naming the content, securing it instead of the channel, and retrieving the content by its names and not the IP address of the machine that hosts the data. NDN has lots of benefits for large scale scientific data as it has very expressive naming support, access control, and enhanced delivery performance compare to traditional IP based networking. NDN supports in-network caching that highly enhances the performance of NDN by reducing latency and network congestion through retrieving popular contents from nearby caches. Considering its importance, we in this work investigate the performance of in-network caching for large scale scientific data. We infer from our simulation results that the optimal cache size and the percentage of the cache hit depend on multiple parameters i.e. traffic pattern, traffic load, and cache replacement policies as proved with simulation results.

**Key Words** : Named-Data Networking (NDN), In-network caching, Future Internet, Caching Replacement Policies

## Ⅰ. Introduction

Information-centric Networking (ICN) is one of the many proposed future Internet architectures which has gained much attention from industry and academia. NDN is the most active instance of ICN where the content/data is retrieved through its name and not the IP address of the host machine unlike traditional IP architecture[6]. NDN is supposed to be more secure than traditional Internet models where the focus is in securing the channel of communication whereas NDN secures the data itself instead of the channel and provides a detailed access control rules[5]. NDN supports in-network caching unlike IP networking. The NDN router's content store (CS) cache popular contents for future use. This considerably enhances NDN performance as it reduces the latency of retrieving the require contents and also help alleviates network congestion by restricting content request (Interest) and data provision to a certain network segment. However, the performance of cache size allocation in NDN routers is dependent upon many factors including traffic pattern, cache placement and replacement policies[4].

The performance parameter for caching performance validation is cache hit ratio or percentage of cache hit. Cache hit mean the number of times the data was requested and it was found in an in-network cache while cache miss is when the data was requested and was not found in any of the in-network cache. Cache performance depends on

---

◆ First Author : University of Science and Technology (UST), Daejeon, South Korea, 학생회원
° Corresponding Author : Korea Institute of Science and Technology Information (KISTI), hklim@kisti.re.kr, 정회원
논문번호 : 201908-150-B-RU, Received August 9, 2019; Revised August 23, 2019; Accepted August 23, 2019

various parameters. The contribution of this paper is that we investigate the dependence of cache performance and optimal cache size on the different traffic pattern, cache replacement policies, and the traffic load. This study provides an insight into the caching performance with different factors in NDN for large scale climate data.

## Ⅱ. Related Work

Where and how to cache and manage large-scale scientific data in distributed memories with limited size forms a variety of research issues[4,7]. Among these issues, a simulation study that showed the caching and aggregation effects on NDN for large-scale climate data was presented[4]. The study shows the dependence of cache size on traffic volume and traffic patterns using a real traffic traces. One result in the work shows that even a 1 GB cache size in edge NDN nodes can provide a magnificent reduction in server hits and network traffic. Since cache size depended on the traffic volume and traffic pattern, the results for the optimal cache size were different in different week traces[4]. In [10], the authors have discussed the dependence of cache size on network topology, request pattern and content popularity, however, they did not show the optimal cache size in different scenarios and network parameters. Authors in [11], show cache hit ratio as a function of cache size under different request pattern, however their goal is to validate the improved performance of their proposed caching scheme compare to others. This paper does not focus on the effect of different network parameters on optimal cache size. The work in [12] shows the cache hit ratio with 2 different topologies and varying the number of caches in the network. The authors have showed that not all nodes must have cache for improved performance and it depends on the network topology to decide where and how many nodes in the network must have cache for performance enhancement. The work in [9], shows the dependence of optimal cache size on catalog size and varying the number of consumers retrieving those files. The authors showed that cache hit ratio was high at low catalog and was low at higher catalog size. Similarly, the hit ratio increased with increasing the number of consumers because of the possible more requests for the same content. Our work shows the dependence of cache size on traffic pattern, cache replacement policies and traffic load.

## Ⅲ. Cache replacement policies

We have used the very general cache replacement policies i.e. FIFO, LRU, LFU and Random. FIFO stands for “first in first out” which means, this cache replacement policies replaces the content that was inserted first like a queue. LRU is least recently used that replaces the content which was used least recently in time while LFU replaces content that was used least frequently compared to other contents in the cache. Random policy randomly selects any content and replaces it. The following figure 1.0 briefly explains these cache replacement policies. In Random policy when a new content 4 arrives, it will just randomly replace any content with 4 as shown in figure below. FIFO will replace content 7 as content 7 is at the head of the cache, which mean, it was inserted first into the cache. LRU replaces content 7 as it was used least recently as shown by

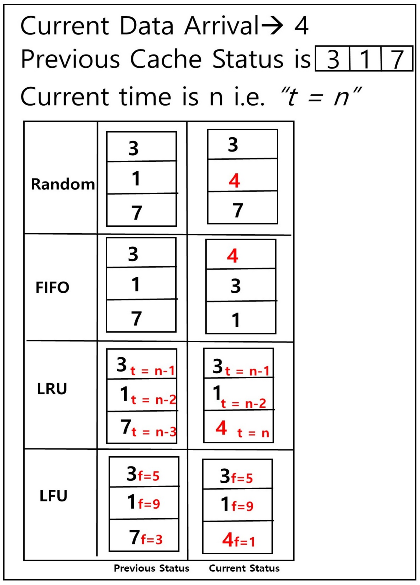

Fig. 1. Cache replacement policies

the time (t = n-3) that means it was used least recently while content 3 (t = n-1) was used most recently followed by content 1 (t = n-2). LFU increment the frequency of use of each content and replaces the content that is used least frequently. In the figure below, content 3 is used 5 times, content 1 is used 9 times while content 7 is used only 3 times; therefore, content 7 will be replaced by LFU upon the arrival of new content 4.

## Ⅳ. Simulation environment and Results

### 4.1 Simulation environment

We have simulated the following network topology as shown in figure 2. we have 4 consumers attached to router 1 and router 2. The cache is located at router 3 only. The data is retrieved from a producer reside at a long distance network as shown in the figure below.

To investigate the caching performance and optimal cache size for large scale climate data, we have simulated cache hit ratio as a function of different cache replacement policies under uniformly and exponentially distributed Interest traffic pattern shown in Figure. 3 and Figure. 4. It is already showed that the content popularity distribution follows a Zipf-like distribution[1]. The a parameter of Zipf-law determines the skewness of distribution and is related to the user's request behavior. A higher value of a determines that the requests are more concentrated on a specific content that means that some specific content is more popular. Different applications have different skewness parameter depending on the nature of the application. It is claimed in [2] that the content popularity of a user generated content service follows a pattern with a value approximately equal to 0.85 while the value of vary between 0.65 and 1 for a video on demand services (VOD)[3]. For big science application the skewness parameter value 1.15 best fits to the user request pattern[4]. In our simulation environment, each consumer (Consumer 1, 2, 3, and 4) generates Interest packets at a rate of 200 Interests per second to retrieve the climate data file of 1.35 GB size at the Producer 1. The producer generates one data packet (i.e., segment) with the size of 8.7 kB (i.e., chunk size) every Interest packet. We use two different Interest traffic patterns, which are uniformly and exponentially distributed traffic models.

### 4.2 Simulation results

We have repeated the simulation for many times and took the average value for each results shown below. We observed the optimal cache size for the two traffic models using well-known cache replacement policies i.e. LRU (Least Recently Used), LFU (Least Frequently Used), FIFO (First in First Out) and Random[8]. Figure. 3 shows that the optimal cache size for FIFO under uniformly distributed traffic is 300 chunks with cache hit rate of 69.38%, For LRU its 500 chunks and cache hit ratio of 69.38%, for Random 15000 chunk with

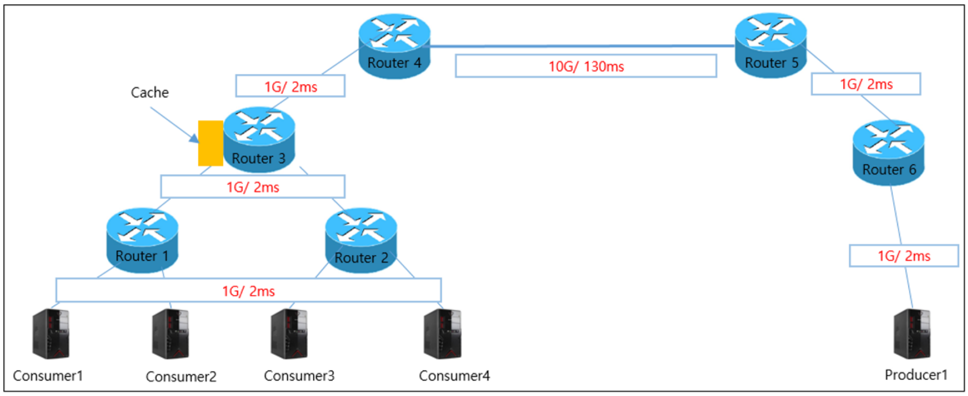

Fig. 2. Network topology

cache hit ratio of 69.38% and for LFU the cache hit ratio is 69.3% achieved with an optimal cache size of 20000 chunks.

The same tests were performed for the exponentially distributed traffic pattern. The results in Figure. 4 shows the optimal cache size observed for FIFO is 500 chunks, for LRU 800, for Random 18000, and for LFU 20000 chunks with cache hit ratio of 74.69% in all cache replacement policies. In both traffic patterns, all consumer request segments sequentially at a different random time. So FIFO performs better because FIFO replaces one segment from the tail of the cache which is no longer needed by any of the consumers as it is already served to the consumers before. LRU also replaces one segment from the tail as segment cached at the tail is least recently used one and also there is more possibility that the segment may no longer be needed by some or any of the consumers. Random replaces any segment randomly so its performance is not good. LFU performs the worst because each segment does not have any popularity and as a result, it replaces the segment at the head of the cache that is not yet served to all the consumers (i.e., the latest cached segment at the head of the cache). The observations attest our arguments about the dependence of optimal cache size on traffic pattern and cache replacement policy.

We also showed that optimal cache size depends upon the traffic volume in the network. Figure 5

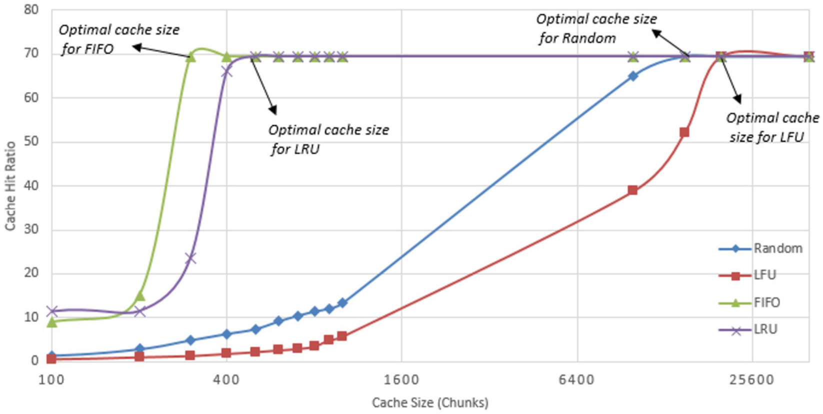

Fig. 3. Cache hit ratio as a function of cache size for different cache replacement policies under uniformly distributed Interest traffic pattern

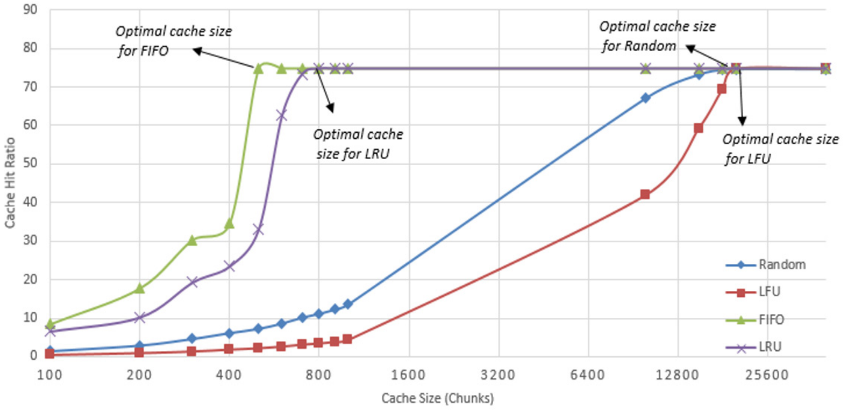

Fig. 4. Cache hit ratio as a function of cache size for different cache replacement policies under exponentially distributed Interest traffic pattern

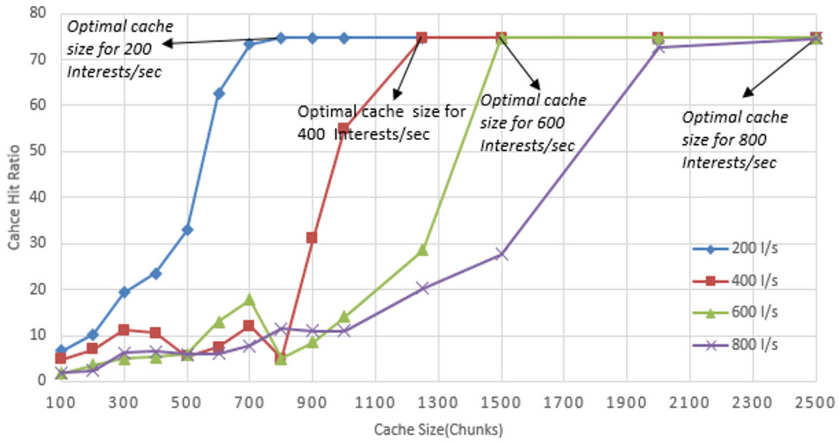

Fig. 5. Cache hit ratio as a function of cache size for different traffic volumes (LRU was used as cache replacement policy).

shows optimal cache size under different traffic volumes (i.e. 200 Interests/sec, 400, 600 and 800 interests/sec). In this experiment, LRU is used as cache replacement policy with exponentially distributed traffic, which is the most widely used cache replacement policy. The result shows that optimal cache size increases with the increase of traffic volume in the network.

## Ⅴ. Conclusion

In this paper, we have investigated in-network caching in NDN for large scale scientific data. We showed that the optimal cache size depends on the traffic pattern, cache replacement policy and traffic load. The study presented in this work will provide an insight into NDN in-network caching to the researchers working in this area to analyze the cache behavior in their experiments, and simulations, and to assign a proper cache size to their particular scenario keeping in mind the takeaways from this study.

## References

[1] L. Breslau, P. Cao, L. Fan, G. Phillips, and S. Shenker, "Web caching and Zipf-like distributions: Evidence and implications," in *Proc. 18th Annu. Joint Conf. IEEE Comput. Commun. Soc. (INFOCOM)*, vol. 1, pp. 126-134, Mar. 1999.

[2] Y. Carlinet, B. Kauffmann, P. Olivier, and A. Simonian, "Trace-based analysis for caching multimedia services," 2011.

[3] H. Yu, D. Zheng, B. Y. Zhao, and W. Zheng, "Understanding user behavior in large-scale video-on-demand systems," *ACM SIGOPS Oper. Syst. Rev.*, vol. 40, no. 4, pp. 333-344, 2006.

[4] S. Shannigrahi, C. Fan, and C. Papadopoulos, "Request aggregation, caching, and forwarding strategies for improving large climate data distribution with NDN: A case study," in *Proc. Inf. Centric Netw. 2017*, pp. 54-65, 2017.

[5] Z. Zhang, Y. Yu, A. Afanasyev, J. Burke, and L. Zhang, "NAC: Name-based access control in named data networking," in *Proc. 4th ACM Conf. Inf. Centric Netw.*, pp. 186-187, Sep. 2017,

[6] L. Zhang, et al., "Named data networking," *ACM SIGCOMM Comput. Commun. Rev.*, vol. 44, pp. 66-73. Jul. 2014.

[7] E. Yeh, et al., "VIP: A framework for joint dynamic forwarding and caching in named data networks," in *Proc. ICN'14*, pp. 117-126, Paris, France, Sep. 2014.

[8] V. Martina, M. Garetto, and E. Leonardi, "A unified approach to the performance analysis of caching Systems," *IEEE Int. Conf. Comput.*

*Commun.*, pp. 2040-2048, 2014.

[9] H. Lim, A. Ni, D. Kim, Y. B. Ko, S. Shannigrahi, and C. Papadopoulos, "NDN construction for big science: Lessons learned from establishing a testbed," *IEEE Network*, vol. 32, no. 6, pp. 124-136, 2018.

[10] 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 *Proc. ACM/IEEE Symp. Architectures for Netw. and Commun. Syst.*, pp. 15-26, Oct. 2012.

[11] M. Rezazad and Y. C. Tay, "CCndnS: A strategy for spreading content and decoupling ndn caches," in *IEEE 2015 IFIP Netw.*, pp. 1-9, Toulouse, France, May 2015.

[12] E. Aubry, T. Silverston, and I. Chrisment, "Green growth in NDN: Deployment of content stores," in *2016 IEEE Int. Symp. Local and Metropolitan Area Netw. (LANMAN)*, pp. 1-6, Jun. 2016.

### Inayat Ali

Inayat Ali is a graduate researcher in Korea Institute of Science and Technology Information (KISTI), campus of the University of Science and Technology (UST), Daejeon, South Korea. Prior to joining KISTI, he has completed his undergraduate studies from COMSATS University Islamabad, Abbottabad campus. His research interest includes Information-Centric Networking, Mobile Edge computing, and privacy and security in IoT.

[ORCID:0000-0002-8643-3452]

### 임헌국 (Huhnkuk Lim)

1999년 2월 : 항공대학교 전자공학과 학사
2001년 2월 : 광주과학기술원 정보통신공학과 석사
2006년 2월 : 광주과학기술원 정보통신공학과 박사
2006년 3월~현재 : 한국과학술정보연구원 책임연구원
2010년 9월~현재 : 과학기술연합대학원대학교 (UST) 그리드 및 슈퍼컴퓨팅 전공 부교수
<관심분야> 통신 컴퓨팅 융합, 미래인터넷, 정보중심네트워킹 (ICN), 엣지 네트워크 시스템, Named Data Networking (NDN), 머신러닝/딥러닝, Connected car.
[ORCID:0000-0002-8032-1597]