International Journal of Advanced Science and Technology
Vol.92 (2016), pp.9-16
http://dx.doi.org/10.14257/ijast.2016.92.02

A Cache Partition Policy of CCN based on Content Popularity

Pu Gong and HuanYu Wu

School of Computer Science and Communication Engineering,
Jiangsu University, China
gp15262908244@qq.com

Abstract

As a representative network architecture of future network, CCN has attracted more and more attention from academia and industry. Among many components of CCN, caching policy is the key feature, which directly influences CCN's performance. In this paper, an interesting topic is explored, i.e. whether the network performance will be promoted if popular and non-popular contents are cached respectively. A cache partition policy based on content popularity is proposed with the theoretical expression derived and the Matlab calculation results provided to verify its performance. The numerical results have shown that, the proposed policy is always superior to the commonly used LCE/LRU policy due to the increased characteristic time.

Keywords: *Next-Generation Internet; Content Centric Network; Caching Policy; Cache Partition; Content Popularity*

1. Introduction

With the rapid development of Internet, network users pay more and more attention to the contents network delivers instead of the location they are stored, and the network applications also are shifting to content services gradually. Unfortunately, the traditional IP network architecture is based on host-to-host mode, so it cannot satisfy current Internet development requirements. To solve this problem, some innovative schemes have been proposed in the first half of 2000s, such as CDN, P2P [1]. But these schemes only relieve the shortcomings of traditional network architectures. Since 2007, European and American researchers have started several research projects on next generation Internet architecture, including DONA[2](Data-Oriented Network Architecture) proposed by UC Berkeley RAD lab, 4WARD[3] by European Union FP7, PSIRP[4] (The Publish-Subscribe Internet Routing Paradigm) and CCN[5-6](Content-Centric Networking) by Palo Alto Research Center, and NDN (Named Data Networking) by NSF Future Internet Architecture(FIA). Without exceptions, these projects all adopt content-centric idea to design network architecture. So currently, CCN has become a representative orientation and hotspot of next generation Internet research.

CCN achieves the goal of separating of content from its location by specifying unique name to each piece of content. Through certain caching policy and naming routing mechanism, the requested content can be obtained from nearby network nodes, without the need of forwarding the request up to the source server. Caching policy is the key component of CCN, and a well-designed caching policy can effectively reduce the request response time and alleviate network congestion. Accordingly, caching policy has been a focus of CCN research.

Two kinds of caching policies are studied in CCN, cache replacement policy and cache decision policy. The former one concerns the replacement of selected cached content with newly-arrived content and the latter concerns the decision over whether to cache newly-arrived content. As far as decision policy is concerned, the default LCE policy caches new contents everywhere. It is simple but with the cost of increasing content redundancy in the

ISSN: 2005-4238 IJAST

network[7]. So researchers proposed a series of improved decision policies to enhance network performance, including Leave Copy Down (LCD), Move Copy Down (MCD), Leave Copy Probability (LCP) [8], ProbCache Policy[9], WAVE[10], Less for More Policy[11], *etc*. They are designed with two aims, one is to decrease copy redundancy and the other to migrate the popularity contents to the node nearby. Different from cache decision policies, cache replacement policies define how and where the retrieved contents will be stored. Least Recently Used (LRU) is the most common policy in CCN [12]. Other policies include Most Recently Used (MRU) and Most Frequently Used (MFU) in [13]. These policies are simple to be implemented in CCN but they don't utilize the content centric characteristics (such as content popularity). Obviously, only well-cooperative decision and replacement policy can maximize the performance of CCN. In this paper, taking both decision and replacement policy into consideration, we propose a partition caching policy based on content popularity, naming it as PCP for short. PCP tries to balance the stored proportion of popularity contents and non-popularity contents in different network location. On the one hand, PCP deals with popular and non-popular contents respectively, avoiding the interrelationship of popular and non-popular contents in the cache. On the other hand, PCP keeps appropriate balance between caching of these two kinds of content, which comes from the following observation. If the caching policy considers only popular contents, it will lead to network filling with very few highest popular contents and excluding contents of normal and low popularity In this case, the whole network performance will decrease due to the terrible retrieved delay for these less popular contents. This is the important problem which will be solved in PCP.

The reminder of this paper is organized as follows. Section 2 describes the architecture of CCN. Section 3 presents PCP caching policy with its theoretical analysis. Section 4 provides performance evaluation through numerical analysis and comparison between PCP and LRU/LCE. Finally, Section 5 concludes the work of the paper.

2. CCN Description

Regarding each requested object as content is the basic thought of CCN. As a consequence, each content in CCN has been assigned an unique name identification. The naming mechanism of CCN is similar to the URL naming scheme. It adopts hierarchical scheme, for instance "/search.com/music/example.mp3" can be a name of the content in CCN. Then "/search.com" and "/search.com/music" are used as prefix when searching and forwarding.

Two types of CCN packet are used: interest packet and data packet. Interest packet with identification is sent by the requester and CCN nodes in the network will forward this packet based on the identification until it reaches a node which can provide the requested content. Then the content is sent back using the data packet to the requester along the reverse path forwarding to complete the communication process. The requested content will be cached as much as possible in CCN nodes along the reverse path, so it can be quickly provided for the subsequent request of the same content from other users. This is totally different from the traditional IP router, which clears its cache after forwarding completed.

The key structure of a CCN node is composed of Content Store (CS), Pending Interest Table (PIT) and Forwarding Information Base (FIB). FIB is used to forward interest packet toward potential source server. PIT records the received interest packets with their arriving faces, which are being pended for response. CS is same as the buffer memory of IP router but has different replacement policy. Since each IP packet belongs to an independent point-to-point conversation, it has no further utility after being forwarded. Thus IP router clears its buffer immediately after forwarding finishes (this action can be regarded as MRU replacement policy). CCN packets are potentially useful to many other users (*e.g.*, other users may read the same newspaper too). To maximize the probability of

sharing, as well as to minimize downstream latency, CCN caches received data packets as long as possible (*e.g.*, LRU or LFU replacement).

3. PCP Description and Analysis

Based on the description of CCN caching policy design in section one, we can find that a good caching policy should guarantee the performance of both popularity and non-popularity contents. For popularity contents, the policy should try to increase their cache residence time so that user has more chances to fetch these contents at network edge. For non-popular contents, the policy should ensure a suitable caching proportion in network and then avoid fetching these contents from source server. According to the above consideration, we try to design a partition caching policy to meet the requirements of CCN, and then explore the operation results of separating the popular contents from ones. In this section, we first present the definitions of popular and non-popular contents, and then describe our policy-PCP and its theoretical performance analysis. In our setting, we make the following assumptions: (1) the network is L-level tandem architecture with one source server, as shown in Figure 1; (2) the source server provides M contents equally partitioned in K different popularity classes. Each class has $m = M / K$ content files and each content is segmented into σ chunks, σ is the average content size; (3) the cache size of each node is C chunks; (4) the arrival request probability of class k in level i is defined as q_k^i .For the first level, the popularity of requests follows $Zipf$ distribution, so q_k^1 can be expressed as $q_k = c / k^\alpha$, $c > 0$, where α is the concentration level of content popularity.

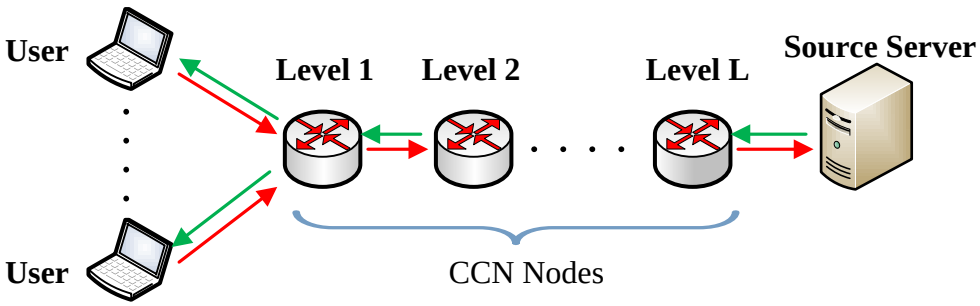


Figure 1. Network Topology: L-level Tandem Architecture

Assume request arrival rate of class k at level i is λ_k^i and Δt is the measure interval for CCN node. The average request number (ARN) of level *i* can be defined as

$$\frac{1}{K}\sum_{k=1}^{K}\lambda_k^i \Delta t = ARN_i \tag{1}$$

In general, the request arrival process of class k is modeled as a Poisson Process of intensity λ_k^i .

During Δt , if the number of arriving requests from class k is greater than ARN_i , this class can be regarded as popularity content; otherwise, it should be regarded as non-popularity content. The classification condition is given by

$$\begin{cases} \lambda_k^i \Delta t > ARN_i & \Rightarrow \quad popularity \\ \lambda_k^i \Delta t < ARN_i & \Rightarrow \quad unpopularity \end{cases} \tag{2}$$

Policy Description: In order to balance the caching proportion of popular and non-popular contents in the CCN node of i th level, we propose to divide up cache into two areas by a ratio of r_i, which is defined as expression (3). The area $C \cdot r_i$ is used for popular contents and the rest part of cache $C \cdot (1-r_i)$ for non-popular contents. A fetched content will be cached in the corresponding area according to its popularity. When the storage is full, the LRU will be used to make space for the new content.

$$r_i = \sum_{j=1}^{K_1} q_j^i \tag{3}$$

Where r_i is the partition ratio for the node in i th level; K_1 is the maximum class of popularity classes; q_j^i is the request probability of class j in level i .Using this adaptive and adjustable parameter, the popular contents will to cached in the nodes at network edge. At the same time, the non-popular contents also get more chances to be cached in network. Obviously our policy is a compromise for these two different kinds of contents, maybe the performance of the highest popularity class will decrease a little, but the entire performance of network will be guaranteed. In the rest part of this section, we will do some theoretical analysis to measure the network performance.

In this part, an important parameter is adopted to evaluate the policy performance, namely, average hit probability of network. We use π_k^i to represent the average hit probability of class k in level i , π_k to denote the total hit probability of class k in network ($1-\pi_k$ means the average hit probability of source server), π to denote the average hit probability of network.

(1) First Level

We first analyze the average hit probability for the first level. Because the replacement policy of each partition is still LRU, we import characteristic time to indicate cache feature [8].

Characteristic time: it denotes the average residence time for a chunk in the cache. According to LRU, a chunk will be moved to the head of cache queue when it is hit and it will be moved backward if there are other chunks' hit events until it is evicted from the cache. So characteristic time represents the average moving time from cache queue's head to the tail for a content, just as shown in Figure 2. We use τ_k^i to represent the characteristic time of class k in level i .

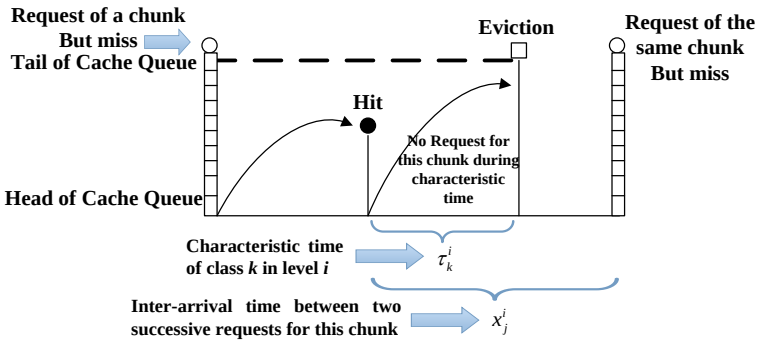


Figure 2. Characteristic Time of Cache

According to its definition, we can obtain the following equation to solve τ_k^1 (we invite reader to [4] for the derivation of equation (4)).

$$\begin{cases} m\sigma \sum_{j=1, j\neq k}^{K_1} \pi_j^i(\tau_k^i) = C \cdot r, \ popularity\ class \\ m\sigma \sum_{j=K_1, j\neq k}^{K} \pi_j^i(\tau_k^i) = C \cdot (1-r_i), unpopularity\ class, \quad i=1 \\ \pi_j^i(\tau_k^i) = P\{x_j^i < \tau_k^i\} = 1 - e^{-\frac{\lambda_j^i}{m}\tau_k^i} \end{cases} \tag{4}$$

where $\pi_j^1(\tau_k^1)$ means the average hit probability for every chunk of class j in level 1 during the interval τ_k^1 ; $m\sigma$ is the total chunk number of one certain class; $m\sigma\pi_j^1(\tau_k^1)$ means the average hit number for class j during the time interval τ_k^1. If the total chunk hit number of all class (except for class k) equals to the cache partition size, a chunk of class k will move from cache head to cache tail according to LRU. So τ_k^1 is the characteristic time of class k ; x_j^1 is the inter-arrival time between two successive requests for a specific chunk of class j and λ_j^1 / m is the arrival rate of this chunk. Because arrival process obeys Poisson Process, the average hit probability should be a distribution function of negative exponential distribution. If $x_j^1 < \tau_k^1$, the successive requested chunk will be hit in this level.

Then the average hit probability of class k in level 1 can be expressed as follows.

$$\pi_k^i = 1 - e^{-\frac{\lambda_k^i}{m}\tau_k^i}, i=1 \tag{5}$$

(2) Network of Caches

Similar to the analysis of first level, we can further obtain average hit probability of class k in level i using equation (4) and (5). In view of requests which has been satisfied in level $i-1$, we can use expression (6) to show the calculation of the request arrival rate for class k in level i.

$$\lambda_k^i = (1-\pi_k^{i-1}) \cdot \lambda_k^{i-1}, i = 2, \cdots, L \tag{6}$$

So π_k can be written as

$$\pi_k = \pi_k^1 + \sum_{i=2}^{L} (1-\pi_k^{i-1})\pi_k^i \tag{7}$$

The average hit probability of network is

$$\pi = \sum_{k=1}^{K} q_k \cdot \pi_k \tag{8}$$

4. Numerical Analysis Results

In order to assess the performance of PCP, we design a test scenario and calculate the above equations for evaluation. We use Matlab as numerical analysis tool and adopt the average hit probability of network as evaluating indicator.

Refer to the simulation parameters from literature [13], we set the network is a triple-level tandem architecture, the source server provides a total of M= 40,000 content files, these files were divided into K = 200 classes, each class had m = 200 content files, and average size of each file equals $10MB$. We further set the average chunk size $\sigma = 10kB$, so each file can be divided into 1000 chunks. Because the requests for contents are generated by Poisson process, we assume the request rate at the first level is $\lambda_1 = 40contents/\sec ond$.

Considering that the design of PCP is based on the contents popularity, the Zipf parameter α obviously is a key factor affecting network performance. Another important factor should be tested is always cache size of node(C) in that cache is the core feature of CCN. Under the above considerations, we select several typical values of these two factors to assess PCP's performance. The detail parameter choices are as follows.

(1) $\alpha = 0.8, \alpha = 1.2, \alpha = 1.5$

(2) $C = 10GB$ (10^6 chunks), $C = 20GB$ (2×10^6 chunks)

First we calculate the popular-and-non-popular-class range according to equations (1)(2), as shown in Table 1. The classification results show that (1) With the increasing of α, the content requests become increasingly concentrated in first several classes; (2)Because the partition ratio of PCP is based on the proportion of request for popular content, the partition ratio increases as α increases and the data in Table 1 prove this viewpoint. Note that the partition ratios of different level are not same. The reason is that the amount of requests for every class is the unsatisfied requests from downstream nodes.

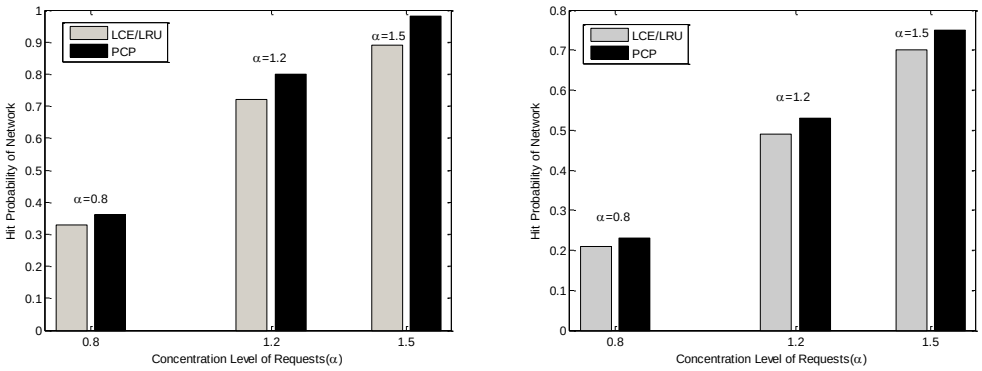
Table 1. Classification Results

	$\alpha = 0.8$	$\alpha = 1.2$	$\alpha = 1.5$
Class range of popular contents	1~42	1~26	1~18
Class range of non-popular contents	43~200	27~200	19~200
Partition ratio of 1st level: r_1	0.6148	0.7761	0.869

Figure 3 shows the network performance comparison of PCP and LCE/LRU policy when cache size and α change. We can clearly find that the PCP is superior to routine LCE/LRE policy in all cases we selected. Why can partition design promote the total hit probability? The reason is easy to understand. When we separately deal with popular/non-popular contents, the arrival rate for popularity partition will decrease. Although the size of popular partition is also smaller (only a part of original cache), the characteristic time for popular classes increases a little, and then hit probability raises accordingly. It has been theorized that excluding the requests for non-popular contents leads to shorter cache update interval. This is the key advantage of partition policy design. Similarly, for the partition of non-popular classes, after routers excluding the requests from popular classes, the cache update time also becomes shorter (the extent to which characteristic time increases is less than that of popular classes), so the hit probability of non-popular contents will also increase a little. Although we change the Zipf parameter α and cache size in our test, we get the same conclusion.

There is another thing worthy of some explanation. In ordinary CCN research papers, researchers always use average hit probability of first several classes to illustrate the performance of popularity classes, for example k varies from 1 to 10. Because we care

about both popular and non-popular classes, we select average hit probability of network as evaluating indicator to show the overall performance in our analysis.



(1)C=20GB (2)C=10GB

Figure 3. Network Hit Probability Comparison between PCP &LCE/LRU

5. Conclusions

In order to enhance CCN cache utilization and improve its performance, this paper proposes a partition policy based on content popularity, provides its theoretical analysis, and then do corresponding numerical evaluation using Matlab.

The basic design of this policy is to categorize all contents into popular and non-popular classes, further divide cache into two partitions for popular and non-popular classes, then implement caching action respectively. Numerical results show that the performance of CCN can be improved using this partition policy. Although the performance promotion of our policy is not remarkable, it indeed improves network performance compared with traditional policy. So the idea of partition or dealing with popular/non-popular classes separately is still worthy of reference in future CCN caching policy research.

In our opinion, how to design the optimized partition ratio for different network level or for the overall network cache should be a valuable research problem in future research.

Acknowledgments

This research is financially supported by the Innovation Program for Graduate Student of Jiangsu Province (Grant No. 2014-467), Jiangsu Province Innovation Training Program of University Student under Grant No. 201610299062Y and Innovation Practice Fund of Industry Center of Jiangsu University under Grant No.2014-23.

References

[1] G. Pallis and A. Vakali. “Insight and perspectives for content delivery networks”, Communications of the ACM, **(2006)**, pp. 101-106.
[2] M. Teemu Chawla Koponen, B. G. Chun, A. Ermolinskiy, K. H. Kim and S. Shenker, “A data-oriented (and beyond) network architecture”, Acm Sigcomm Computer Communication Review, vol. 37, no. 4, **(2007)**, pp. 181-192.
[3] J. Pan, S. Paul and R. Jain, “A survey of the research on future internet architectures”, Communications Magazine IEEE, vol. 49, no. 7, **(2011)**, pp. 26-36.
[4] N. Fotiou, P. Nikander, D. Trossen and G. C. Polyzos, “Developing Information Networking Further: From PSIRP to PURSUIT”, Lecture Notes of the Institute for Computer Sciences Social Informatics & Telecommunications Engineering, **(2012)**, pp. 1-13.
[5] V. Jacobson, D. K. Smetters, J. D. Thornton, M. F. Plass, N. H. Briggs and R. L. Braynard, “Networking named content”, In CoNEXT ’09: Proceedings of the 5th international conference on Emerging networking experiments and technologies, **(2009)**, pp. 1-12.

[6] L. Zhang, D. Estrin, J. Burke, V. Jacobson, T. D. K. Jim and S. B. Zhang, "Named Data Networking (NDN) Project NDN-0001", Acm Sigcomm Computer Communication Review, vol. 44, no. 3, **(2010)**, pp. 66-73.
[7] K. Katsaros, G. Xylomenos and G. C. Polyzos, "Multicache: An Incrementally Deployable Overlay Architecture for Information-Centric Networking", INFOCOM IEEE Conference on Computer Communications Workshops, **(2010)**, pp. 1-5.
[8] N. Laoutaris, C. Hao Nikolaos and I. Stavrakakis, "The LCD interconnection of LRU caches and its analysis", Performance Evaluation, vol. 63, no. 7, **(2006)**, pp. 609-634.
[9] I. Psaras, W. K. Chai and G. Pavlou, "Probabilistic in-network caching for information-centric networks", In Proceedings of the second edition of the ICN workshop on Information-centric networking, ACM, **(2012)** August, pp. 55-60.
[10] K. Cho, M. Lee, K. Park, T. T. Kwon, Y. Choi and S. Pack, "WAVE: Popularity-based and collaborative in-network caching for content-oriented networks", Computer Communications Workshops (INFOCOM WKSHPS), 2012 IEEE Conference on IEEE, **(2012)**, pp. 316-321.
[11] D. He, G. Pavlou, W. Koong Chai and I. Psaras, "Cache less for more in information-centric networks", In Proceedings of the Ifip-tc6 Networking Conference, DOI:10.1007/978-3-642-30045-5_3, vol. 36, **(2012)**, pp. 27-40.
[12] J. Choi, J. Han, E. Cho, T. Taekyoung Kwon and Y. Choi, "A Survey on content-oriented networking for efficient content delivery", Communications Magazine IEEE, vol. 49, no. 3, **(2011)**, pp. 121-127.
[13] G. Carofiglio, M. Gallo, L. Muscariello and D. Perino, "Modeling data transfer in content-centric networking", Teletraffic Congress (ITC), 2011 23rd International IEEE, **(2011)**, pp. 111-118.

Authors

Pu Gong is a graduate student of Electronics and Communications Engineering in Jiangsu University, China, since 2013. His research interests include information-centric networking, content-centric networking, and mobile application development.

HuanYu Wu is an undergraduate student of Communication Engineering in Jiangsu University, China, since 2014. His research interest is future network, including information centric networking (ICN) and software defined networking(SDN).

