Impact of Item Popularity and Chunk Popularity in CCN Caching Management

Zhi Liu†, Mianxiong Dong⋆, Bo Gu*, Cheng Zhang†, Yusheng Ji‡, Yoshiaki Tanaka†
† Waseda University, Tokyo, Japan 169-8555
⋆ Muroran Institute of Technology, Hokkaido, Japan 050-8585
* Kogakuin University, Tokyo, Japan, 192-9915
‡ National Institute of Informatics, Tokyo, Japan 101-8430
Email: {liuzhi@aoni.waseda.jp, mx.dong@csse.muroran-it.ac.jp,
bo.gu@cc.kogakuin.ac.jp, cheng.zhang@akane.waseda.jp, kei@nii.ac.jp, ytanaka@waseda.jp}

***Abstract*—*Content Centric Network* (CCN) has become a heated research topic recently, as it is proposed as an alternative of the future network. The routers in CCN have the caching abilities and the caching strategies affect the system performance greatly. Each content in CCN is associated with a popularity, which is determined by the corresponding requested times. Popularity-aware caching scheme caches the popular content close to users and can lead to better caching performance in terms of smaller average transmission hops traveled. Content popularity significantly affects the overall system performance, and the content size is not considered during the content level popularity (i.e. item popularity) calculation. In this paper, we study the impact of the item popularity and chunk popularity in CCN, where the chunk popularity is the normalized item popularity considering the content size. Extensive simulations are conducted and the simulation results show the advantages and disadvantages of each scheme. A new popularity calculation method is proposed to perform the tradeoff between the item popularity and chunk popularity.**

***Index Terms*—content centric network (CCN), ICN, caching, item popularity, chunk popularity**

I. Introduction

Content Centric Network (CCN) [1,2] is proposed as an alternative of the current TCP/IP-based network. Due to CCN's good performance, it has recently become a heated research topic in both academia and industry. Different from the TCP/IP-based network, CCN emphasizes 'what' instead of 'where', which makes content a primitive in CCN. Data itself is a name in CCN and can be requested directly at the network level. This means IP addresses and DNS are not necessary. Anybody who has the requested data can answer the data request. Moreover, the authentication and security are conducted directly to the data instead of securing connections the data traverses in traditional networks.

CCN's routers have caching abilities and this could help improve the network to a great extent. By caching contents in routers in advance, the requests of these contents could be satisfied directly by these routers without accessing the remote server. Given the routers are generally located closer to users comparing with the servers, the transmission hops (or total network traffic and transmission time) could be greatly reduced. Meanwhile, this also reduces the server hit rate, which is defined as the ratio of the number of requests satisfied by the server to the total number of data requests.

The inherent problem then becomes how to manage the caching in CCN, which mainly involves *caching decision* and *caching replacement*. The caching decision defines what to cache in each router. And the caching replacement defines which content should be replaced or moved out of the full cache when a to-be-cached content arrives at the router. The caching decision policy and the caching replacement scheme affect the system performance, and there are a branch of researches in this field such as [3,4].

On the other hand, each content has an associated popularity, which is defined based on its requested time and cloud be obtained. As a typical content over the Internet, video is becoming more and more popular nowadays. According to the Cisco Visual Networking Index[1], three-fourths (75%) of the world's mobile data traffic will be video by 2020, up from 55% in 2015, i.e. the mobile video will increase 11-fold between 2015 and 2020. How to stream the video over the network becomes an important problem [5]. It is well studied in literature that some video is requested more often than the rest, and the video distribution cloud be modeled using Zipf [6], Weibull distribution function [7], Gamma distribution, etc. Then given different content has different average requested time, the network benefits by caching the popular contents in the cache closer to users, which could reduce the network traffic to a great extent. There are studies in the popularity-aware caching in

[0]This work is partially supported by JSPS KAKENHI under Grant 15K21599, 16H02817, 16K00117, JSPS A3 Foresight Program.

[1]Cisco Visual Networking Index: Global Mobile Data Traffic Forecast Update, 2015~2020. Tech. rep., 2016.

CCN, e.g., [3,8], which provided good performances over comparison schemes.

The popularity used in these schemes is usually decided by the request times at the content level without taking the content size into consideration. The video size varies significantly due to different lengths, quantization levels, resolutions, etc. And it is non-trivial to judge whether considering the content size leads to better performance or not. In this paper, we study the impact of the item popularity and chunk popularity in CCN, where the chunk popularity is the normalized item popularity considering the content size. Extensive simulations are conducted and the simulation results show both the advantages and disadvantages of the item popularity and chunk popularity. A new popularity is proposed thereafter to help perform the tradeoff between the item popularity and chunk popularity. We expect this paper could provide clues on the future popularity-aware caching scheme design.

The rest of this paper is organized as follows: Section II discusses the existing work about CCN and CCN caching. The system model is introduced in Section III. We show the different popularity calculation methods and the caching scheme in Section IV. The corresponding performance of each popularity is shown in Section V and we conclude this paper in Section VI.

II. Related works

CCN is emerging as an alternative of the future network structure due to its good performance, and it draws great attentions from both academia and industry. Comparing with the traditional TCP/IP-based networks, the disadvantages or bottlenecks introduced by IP addresses could be released. Recently many researches have been conducted from different aspects such as routing [9], security [10], naming [11], mobility [12], etc. to make the CCN more efficient and functional.

Among the many research issues in CCN, CCN caching is one important topic. CCN caching scheme plays an important role in CCN, and pre-caching the contents could greatly reduce the network traffic since the data requests could be satisfied directly by the routers instead of remote server. But the cache size is limited, hence the cache scheme should be carefully designed. There are dozens of researches in this field [13–20]. Specifically, [3] proposed a caching scheme named *Popcach*. This scheme explores the video popularity for the caching scheme design and yields better performance comparing to the benchmark caching decision policies. [14] provided an incentive driven inter-domain caching mechanism for a future network architectures.

These schemes mainly focus on the caching algorithm design and utilize the content popularity if the popularity value could be obtained. In another word, these works do not distinguish the item popularity and the chunk popularity, and the impact of the item popularity and chunk popularity is not investigated. Different from these works, this paper targets to study the impact of the item popularity and chunk popularity in CCN caching. Both the advantages and disadvantages of using item popularity or chunk popularity are shown, and a new popularity is proposed to perform the tradeoff between the item popularity and chunk popularity. This paper is expected to provide some hints on how to select the popularity when the caching scheme is designed. These differences help distinguish our work from the existing schemes.

III. System Overview

CCN is composed by servers/repositories, routers and users/consumers, and Figure 1 shows one typical CCN scenario. Users send data requests to the server and server delivers the requested data to the CCN users. The server along with its following routers and users belong to one domain, i.e. the intra-domain, which is labeled using the dashed line box in Figure 1. This paper focuses on the intra-domain caching and studies the impact of the item popularity and chunk popularity in the CCN intra-domain caching.

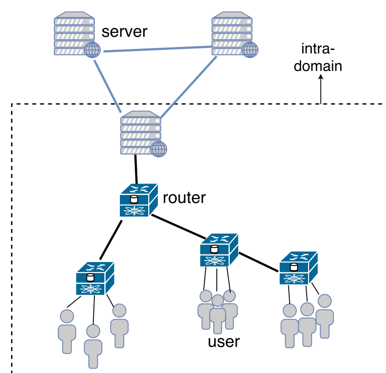


Fig. 1: Illustration of one typical CCN.

In CCN, routers can buffer data. Content store (CS) of CCN routers plays a role of a buffer memory. The CCN caching involves 1) caching decision and 2) caching replacement, which affect the caching efficiency and the overall network performance. Specifically, the router can answer the data request directly if the router has the requested content in the cache, and this means user's data request could be satisfied without asking the server's help. If the content request can be satisfied at router level i, i.e. router i has the requested content, the request will be directly satisfied by the router. Otherwise, the request will be forwarded to the router at level $i+1$. Here the router level is defined based on the distance from users, and the router directly connected with user is level 1 router. The same router may have different levels based on the heterogenous user locations, and how to decide the router level in mesh topology is difficult. This paper focuses on studying the impact of the item popularity and chunk popularity, and we assume the levels are known. This is actually reasonable if we take a look at the last miles of the network, where the network topology is known and router levels are clear.

Hence level 1 router can know all the request information and level i router has the request information except the requests that have been stratified by lower level routers including routers with level $1, ...i$. Then the content popularity could be obtained by each router for the popularity-based caching design. If all the routers along the path between the user and the server do not have the requested content, the request will be satisfied by the server.

Cascade network topology and *Binary tree topology* as shown in figure 2 and figure 3, are two typical network topologies. From the illustrations of these two network topologies, we can observe that there are multiple routers existed along the transmission path between server and users. The routers are numbered based on the corresponding distance from users. The router closest to users are denoted as level 1 router. In Fig. 2 and Fig. 3, level 5 router and level 3 router are the furthest from the users, respectively. In this paper, we assume the routers are marked with router levels and ith level router has cache size to be x_i chunks..

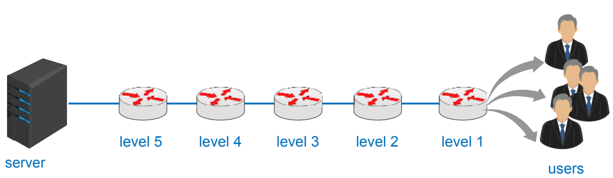


Fig. 2: Cascade network topology illustration, and level 5 router is the most close to the server.

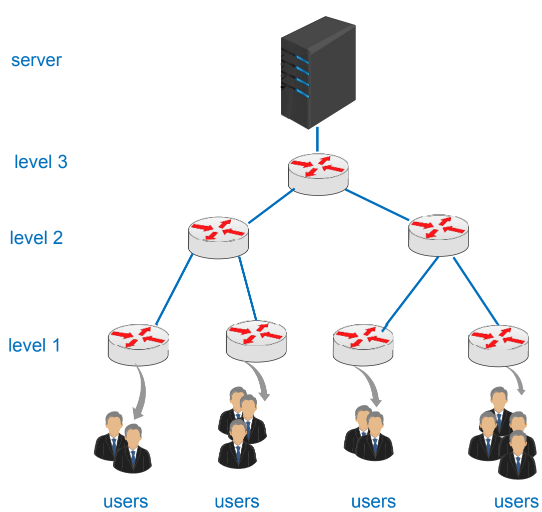


Fig. 3: Binary tree network topology illustration, and level 3 router is the most close to the server.

IV. Caching decision policy and New popularity

This section explains the popularity calculation and the proposed new popularity calculation method. The caching strategy based on the content popularity is then explained at the end of this section.

A. Content popularity

Content popularity is defined as the ratio of the number of content requests to the number of total requests, and different content has different popularity values. Zipf popularity distribution is popularly used to represent the content popularity. Note that other popularity model such as *Weibull* probability function also works in our scheme, by substituting the popularity calculation method used in Zipf by the corresponding popularity calculation methods used in these models. The popularity modeling itself and the content popularity change are out of the scope of this paper, and we use the Zipf popularity distribution to study the impact of the item popularity and chunk popularity. This paper assumes the number of contents is denoted as N and k_i stands for the popularity rank of content i. Higher popularity rank is represented using smaller rank instance number. Based on the Zipf's law, we can calculate the frequency of element i with rank k_i out of a population of N as follows:

$$i_{i,k_i,s,N} = \frac{1/k_i^s}{\sum_{n=1}^{n=N} 1/n^s} \tag{1}$$

Where s denotes the value of the exponent characterizing the distribution and is referred to the skewness of the popularity. Since $i_{i,k_i,s,N}$ does not take the content size into consideration, we name $i_{i,k_i,s,N}$ to be the item popularity.

The content size is also investigated in this paper, and the chunk popularity is defined as

$$c_{i,k_i,s,N} = \frac{1/k_i^s}{r_i \sum_{n=1}^{n=N} 1/n^s} = \frac{i_{i,k_i,s,N}}{r_i} \tag{2}$$

Where r_i is the size of content i. We could observe that the chunk popularity is the normalized version of item popularity $i_{i,k_i,s,N}$, where the video size is taken into consideration. The advantage and disadvantage of item popularity and chunk popularity are shown in Section V in terms of the server hit rate and the network traffic. We also propose a new popularity to perform the tradeoff between the item popularity and the chunk popularity, and it is defined as follows:

$$b_{i,k_i,s,N}^{\alpha} = \frac{1/k_i^s}{r_i^{\alpha} \sum_{n=1}^{n=N} 1/n^s} = \frac{i_{i,k_i,s,N}}{r_i^{\alpha}} \tag{3}$$

In Eq.(3), α is the weight parameter to balance the content size and the item popularity, and $0 \le \alpha \le 1$. We could observer that $b_{i,k_i,s,N}^{\alpha}$ is quite similar with $c_{i,k_i,s,N}$.

If $\alpha = 0$, $b^{\alpha}_{i,k_i,s,N} = i_{i,k_i,s,N}$. If $\alpha = 1$, $b^{\alpha}_{i,k_i,s,N} = c_{i,k_i,s,N}$. The larger the α is, more weight is given to the content size, vice versa. We will show the performance of this new popularity calculation method in Section V.

B. Most popular caching policy

We study the impact of the item popularity, chunk popularity and the new popularity calculation method with the caching policy to be *most popular caching policy* or *MPCP* for short. The MPCP in this paper is divided into three cases: *MPCPitem*, *MPCPchunk* and *MPCPbalance*, which correspond to the item popularity $i_{i,k_i,s,N}$, chunk popularity $c_{i,k_i,s,N}$ and the proposed popularity value $b^{\alpha}_{i,k_i,s,N}$.

MPCPitem is defined as follows: each router selects the content according to the item popularity and the low level router selects the content to cache first. The high popularity items will be selected until the cache is used up. Based on the item popularity $i_{i,k_i,s,N}$, each content is ranked with a resulted rank sequence k_i. Assume ith level router′ CS has a cache size x_i chunks, and the content i′s size is δ_i chunks, we can mathematically calculate the indicator $D^{item}_{i,j}$.

$$D^{item}_{i,j} = \begin{cases} 1 & : if(\delta_i + \sum\limits_{\forall p \in N, k_p < k_i} \delta_p) > \sum\limits_{q=1}^{j-1} x_q \& \sum\limits_{\forall p \in N, k_p \le k_i} \delta_p \le \sum\limits_{q=1}^{j} x_q \\ 0 & : o.w. \end{cases}$$

$D^{item}_{i,j} = 1$ stands for that the jth level router will cache content i, and $D^{item}_{i,j} = 0$ means that jth level router will not cache content i. The value of $D^{item}_{i,j}$ is decided by content i′ popularity rank k_i and routers′ caching abilities, which are determined by the size of the caches and sizes of the contents. The general idea is that the most popular content will be placed in the router nearest to the user. If the router is not empty, the second most popular video will also be placed in this router, otherwise, the video will be placed in the next level router. Then by comparing the total cache capabilities up to level j and the sizes of the contents which have higher popularity rank than k_i, $D^{item}_{i,j}$ could be calculated.

MPCPchunk and *MPCPbalance* are similar with *MPCPitem*. The only difference is that when the content rank is calculated, *MPCPchunk* and *MPCPbalance* use $c_{i,k_i,s,N}$ and $b^{\alpha}_{i,k_i,s,N}$ instead of $i_{i,k_i,s,N}$.

C. Optimality

Now we prove that *MPCitem* is optimal in terms of the transmission hops needed in Cascade topology.

Theorem 4.1: MPCitem is optimal in terms of minimizing the intra-domain network traffic in the CCN Cascade topology.

Proof: Assume *MPCitem* is not optimal, and *MPCitem′* is the optimal caching decision policy. Comparing these two caching schemes, content set Φ chosen by *MPCitem* is substituted by Φ' in *MPCitem′*. Given that Φ caches all the content with highest item popularity, $\exists\ i, j \in \Phi'$, where $k_i > k_j$, but i is stored in the router (server) further from the item j. This is reasonable, otherwise, *MPCitem* becomes the same with *MPCitem′*. We assume the number of hops needed by i, j are d_i, d_j, respectively, and $d_i > d_j$. If $x_i \le x_j$, we change the x_i chunks used by content j in Φ' by content i' chunks. Then the intra-domain traffic introduced by accessing these two contents become $k_i * x_i * d_j + k_j * (x_j - x_i) * d_j + k_j * x_i * d_i$ instead of the traffic $k_i * x_i * d_i + k_j * x_j * d_j$ in *MPCitem′*.$(k_i * x_i * d_j + k_j * (x_j - x_i) * d_j + k_j * x_i * d_i) - (k_i * x_i * d_i + k_j * x_j * d_j) = k_i * x_i * (d_j - d_i) + k_j * x_i * (d_i - d_j) = (k_j - k_i) * x_i * (d_i - d_j)$. Given that $k_i > k_j$ and $d_i > d_j$, $(k_j - k_i) * x_i * (d_i - d_j) < 0$. This means the intra-domain traffic becomes smaller by this substitution since the traffic introduced by accessing other content keeps the same. If $x_i > x_j$, we change all the x_j chunks used for content j in Φ' by content i' chunks. Then the intra-domain traffic of accessing these two contents become $k_i * x_j * d_j + k_i * (x_i - x_j) * d_i + k_j * x_j * d_i$ instead of the traffic reduction $k_i * x_i * d_i + k_j * x_j * d_j$ in *MPCitem′*. $(k_i * x_j * d_j + k_i * (x_i - x_j) * d_i + k_j * x_j * d_i) - (k_i * x_i * d_i + k_j * x_j * d_j) = k_i * x_j * (d_j - d_i) + k_j * x_j * (d_i - d_j) = (k_i - k_j) * x_j * (d_j - d_i)$, since $k_i > k_j$ and $d_i > d_j$, $(k_i - k_j) * x_j * (d_j - d_i) < 0$. Then we can observe that this substitution leads to lower network traffic. Hence, we can claim the *MPCitem′* is not optimal since the intra-domain network traffic could be further reduced by some substitutions. Therefore we can claim that *MPCitem* is optimal in terms of minimizing the intra-domain network traffic in the CCN Cascade topology. ■

Similarly we could also prove that *MPCchunk* leads to the lowest server hit rate in the Cascade topology. The performance of *MPCitem*, *MPCchunk* and *MPCbalance* in other network topologies will be mathematically studied as the future work.

V. Experimental results

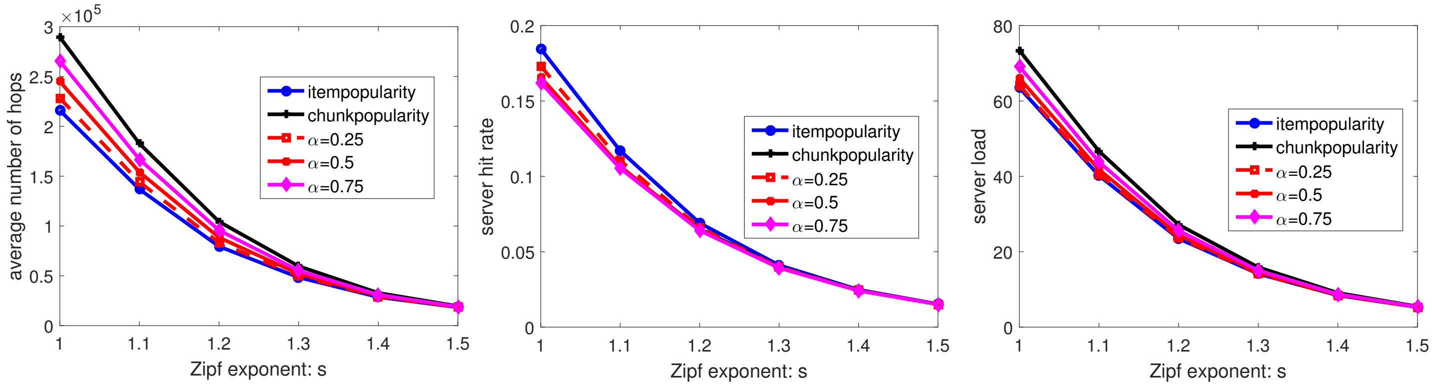
To evaluate the performance of different popularity calculation methods, extensive simulations are conducted besides the mathematical analysis in the previous section. We introduce the simulation setup and the simulation performance in this Section.

A. Simulation setup

TABLE I: Simulation parameters

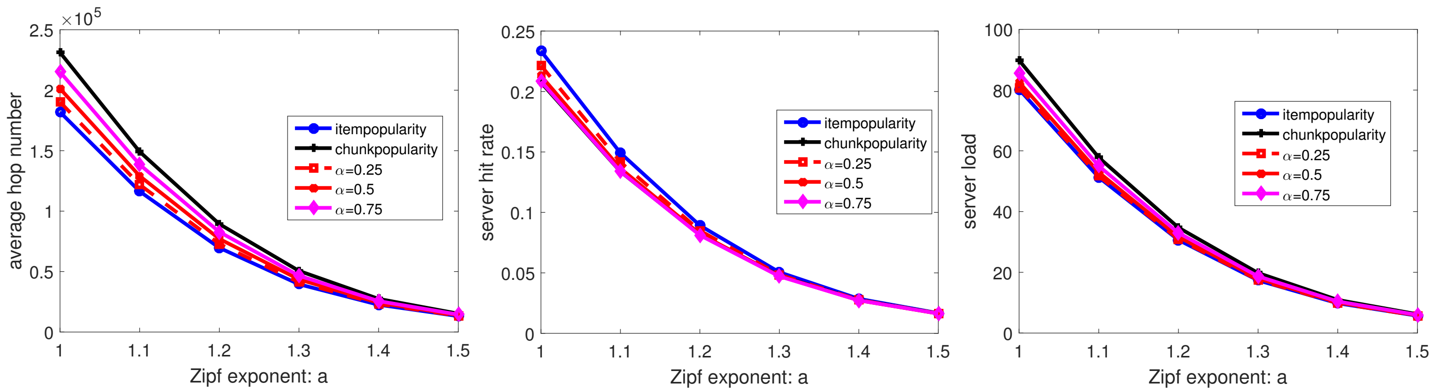
Parameters	Values
total request rate: λ	120 content items/s
# of different content items: M	2×10^4 items
cache size of node 1: X(1)	2×10^5 chunks (2GB)

We conduct a small scale simulation to verify the performance of the different popularity calculation methods. The simulation parameters are shown in TABLE I, where we follow the simulation setup used in [3]. The network topology chosen are the cascade network topology and the binary tree network topology as introduced in Section III. The requests are assumed to be finished within



(a) average transmission hops (b) server hit rates (c) server load

Fig. 4: Impact of different popularity calculation methods with the five level cascade network topology.



(a) average transmission hops (b) server hit rates (c) server load

Fig. 5: Impact of different popularity calculation methods with three level binary tree network topology.

one second in the simulation, which is reasonable due to the large available network bandwidth. We use the video to present the contents in CCN, and the video is assume to be identical but with different popularity ranks and sizes. MPEG test sequence *Kendo*[2] is used and is encoded using JSVM provided by *Fraunhofer HHI*. The resolution of the video is 1024×768, and the frame rate is 30 frame per second (FPS). *Kendo* is encoded at 5 different quantization levels $26, 32, 38, 44, 50$, the resulted video qualities are $44.0dB, 41.1dB, 37.6dB, 34.1dB, 29.2dB$ in terms of PSNR (Peak signal-to-noise ratio), respectively. The content size is assumed to be randomly selected from these five options for simplicity, and the results are averaged with the simulation times to be 100.

B. Simulation results

To study the impact of different popularity calculation methods, we select three different metrics: *server hit rates*, *average number of transmission hops needed*, and *total number of chunks sent by the server (server load)*. The first 1500s are used as the *learning* period, and the cache is updated following the caching policy during this period. The results shown in the figures are the experimental results averaged from 1501s to 2000s.

The simulation performances in Cascade network topology are shown in Fig. 4 including the comparisons in terms of the server hit rate, average number of transmission hops needed and the server load. Note the caching scheme *MPCitem* is optimal for caching in Cascade network topology in terms of the network traffic used as proved in Section IV-C. Specifically, Fig. 4 (a) shows the average number of transmission hops necessary to satisfy the users' data requests. Here we count that one chunk travels from one router to it neighboring router as one hop. From the simulation results, we could observe that the *itempopularity* leads to the best performance among all the popularity calculations as proved in the Section IV-C. And the *chunkpopularity* results in the largest network traffic. This is because *MPCchunk* places some content, which has low item popularity but has smaller content size. *MPCblance* provides different weight to the content size, and its performance is between the *MPCitem* and *MPCchunk*,

[2]http://www.fujii.nuee.nagoya-u.ac.jp/multiview-data/

i.e. *MPCPbalance* is worse than *itempopularity* and better than *chunkpopularity*. The largest the α is, the larger the network transmission hops needed.

From Fig. 4 (b), we can notice that the *MPCchunk* provides the smallest server hit ratio, and the server hit rate of *MPCitem* is the largest. The *MPCblance*'s performance is between *MPCchunk* and *MPCitem*. The reason behind is that the content size is now taken into consideration. Then the content with large size will be given lower priority to help save space and lower popularity content will be given higher priority if its size is small. Then more contents will be cached and the server hit rate is lowered. The proposed *MPCblance* provides different weight to the content size, and hence the performance is between the *MPCitem* and *MPCchunk*. Meanwhile, although the *MPCchunk* can provide the smallest server hit rate, the traffic from the server is the largest among the simulated schemes. *MPCitem* leads to the smallest server load in term of the traffic.

The simulation performance with the binary tree network topology is shown in Fig. 5, where we can observe similar performance as the Cascade network topology as shown in Fig. 4. From the results, we could observe that the *MPCitem* leads to the smallest intra-domain traffic. *MPCchunk* provides the smallest server hit ratio, although the total traffic from the server is larger. The proposed new popularity balances the performance between *MPCitem* and the *MPCchunk*. These different popularity calculation methods could then be selected according to the design objective.

VI. Conclusion

Caching is one fundamental issue in CCN and it greatly affects the system performance. In this paper, we study the impact of the item popularity and chunk popularity in CCN caching. Extensive simulations are conducted and the simulation results show both the advantages and disadvantages of each popularity calculation method. We could find that the item polarity leads to smaller average transmission hops while the chunk popularity provides smaller server hit rate. A new popularity is then proposed to perform the tradeoff between the item popularity and chunk popularity, and we hope this could provide more information for the popularity selection in the future popularity-aware caching design.

References

[1] 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*, ser. CoNEXT '09. New York, NY, USA: ACM, 2009, pp. 1–12.

[2] B. Ahlgren, C. Dannewitz, C. Imbrenda, D. Kutscher, and B. Ohlman, "A survey of information-centric networking," *Communications Magazine, IEEE*, vol. 50, no. 7, pp. 26–36, 2012.

[3] K. Suksomboon, S. Tarnoi, Y. Ji, M. Koibuchi, K. Fukuda, S. Abe, N. Motonori, M. Aoki, S. Urushidani, and S. Yamada, "Popcache: Cache more or less based on content popularity for information-centric networking," in *Local Computer Networks (LCN), 2013 IEEE 38th Conference on*, Oct 2013, pp. 236–243.

[4] I. Psaras, R. G. Clegg, R. Landa, W. K. Chai, and G. Pavlou, "Modelling and evaluation of ccn-caching trees," in *NETWORKING 2011*. Springer, 2011, pp. 78–91.

[5] Z. Liu, G. Cheung, and Y. Ji, "Optimizing distributed source coding for interactive multiview video streaming over lossy networks," *Circuits and Systems for Video Technology, IEEE Transactions on*, vol. 23, no. 10, pp. 1781–1794, 2013.

[6] M. Cha, H. Kwak, P. Rodriguez, Y.-Y. Ahn, and S. Moon, "I tube, you tube, everybody tubes: analyzing the world's largest user generated content video system," in *Proceedings of the 7th ACM SIGCOMM conference on Internet measurement*. ACM, 2007, pp. 1–14.

[7] F. Jiang, Z. Liu, K. Thilakarathnaet, Z. Li, Y. Ji, and A. Seneviratne, "Transfetch: A viewing behavior driven video distribution framework in public transport," in *IEEE LCN*, November 2016.

[8] Z. Liu, M. Dong, B. Gu, C. Zhang, Y. Ji, and Y. Tanaka, "Inter-domain popularity-aware video caching in future internet architecture," in *11th International Conference on Heterogeneous Networking for Quality, Reliability, Security and Robustness (Qshine)*, 2015.

[9] S. Eum, K. Nakauchi, M. Murata, Y. Shoji, and N. Nishinaga, "Catt: potential based routing with content caching for icn," in *Proceedings of the second edition of the ICN workshop on Information-centric networking*. ACM, 2012, pp. 49–54.

[10] C. A. Wood and E. Uzun, "Flexible end-to-end content security in ccn," in *Consumer Communications and Networking Conference (CCNC), 2014 IEEE 11th*. IEEE, 2014, pp. 858–865.

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

[12] D.-h. Kim, J.-h. Kim, Y.-s. Kim, H.-s. Yoon, and I. Yeom, "Mobility support in content centric networks," in *Proceedings of the second edition of the ICN workshop on Information-centric networking*. ACM, 2012, pp. 13–18.

[13] G. Carofiglio, M. Gallo, L. Muscariello, and D. Perino, "Modeling data transfer in content-centric networking," in *Proceedings of the 23rd international teletraffic congress*. International Teletraffic Congress, 2011, pp. 111–118.

[14] K. Suksomboon, Y. Ji, M. Koibuchi, K. Fukuda, S. Abe Nakamura Motonori, M. Aoki, S. Urushidani, and S. Yamada, "On incentive-based inter-domain caching for content delivery in future internet architectures," in *Proceedings of the Asian Internet Engineeering Conference*. ACM, 2012, pp. 1–8.

[15] Y. Li, T. Lin, H. Tang, and P. Sun, "A chunk caching location and searching scheme in content centric networking," in *IEEE ICC*. IEEE, 2012, pp. 2655–2659.

[16] J. Li, H. Wu, B. Liu, J. Lu, Y. Wang, X. Wang, Y. Zhang, and L. Dong, "Popularity-driven coordinated caching in named data networking," in *Proceedings of the eighth ACM/IEEE symposium on Architectures for networking and communications systems*. ACM, 2012, pp. 15–26.

[17] H. Wu, J. Li, T. Pan, and B. Liu, "A novel caching scheme for the backbone of named data networking," in *Communications (ICC), 2013 IEEE International Conference on*. IEEE, 2013, pp. 3634–3638.

[18] D. Rossi, G. Rossini *et al.*, "On sizing ccn content stores by exploiting topological information." in *INFOCOM Workshops*, 2012, pp. 280–285.

[19] Q. Wu, Z. Li, and G. Xie, "Codingcache: multipath-aware ccn cache with network coding," in *Proceedings of the 3rd ACM SIGCOMM workshop on Information-centric networking*. ACM, 2013, pp. 41–42.

[20] Z. Liu, M. Dong, B. Gu, C. Zhang, Y. Ji, and Y. Tanaka, "Fast-start video delivery in future internet architectures with intra-domain caching," *Mobile Networks and Applications*, pp. 1–15, 2016.