A Cluster-based Cache Distribution Scheme in Content-Centric Networking

Mikiya Yoshida
University of Kitakyushu
tori29@net.is.env.kitakyu-u.ac.jp

Yusuke Ito
University of Kitakyushu
wash000i@net.is.env.kitakyu-u.ac.jp

Yurino Sato
National Institute of Technology, Sasebo College
y-sato@sasebo.ac.jp

Hiroyuki Koga
University of Kitakyushu
h.koga@kitakyu-u.ac.jp

ABSTRACT

Content-centric networking (CCN) promises efficient content delivery services with in-network caching. To improve cache efficiency, some approaches that efficiently distribute and obtain various caches over a domain, have been proposed. However, the time needed to obtain content gets longer in larger scale networks. We therefore propose an efficient cluster-based cache distribution scheme to improve the time needed to obtain content as well as cache efficiency. We evaluate the effectiveness of this approach through simulation.

CCS CONCEPTS

• **Networks** → **Network architectures**; **Network performance evaluation**; **Network simulations**;

KEYWORDS

CCN, Clustering, Cache distribution

1 INTRODUCTION

Content-centric networking (CCN), which is designed for content delivery services, has attracted much attention [1]. In CCN, content is segmented into smaller chunks that are stored in content routers (CRs) during forwarding. A user requesting content sends interest packets to the server along the shortest path, called the default path, which is constructed by the forwarding information base (FIB) on each CR. A CR that receives the interest packets sends the requested chunks back to the user instead of the server if it caches them; otherwise, it forwards the interest packets to the server. Namely, users can obtain chunks from nearby CRs if efficiently cached. However, each CR tends to mostly cache highly popular contents because frequently requested chunks are often duplicated. This will degrade cache efficiency in obtaining various contents in CCN.

To improve cache efficiency, some approaches have been proposed [2, 3]. These schemes use a hash function to distribute and obtain various caches to/from all CRs in a domain. Although cache efficiency can be improved by distributing cached chunks over the domain, the time needed to obtain content gets longer because some chunks may be retrieved from distant CRs in larger scale networks.

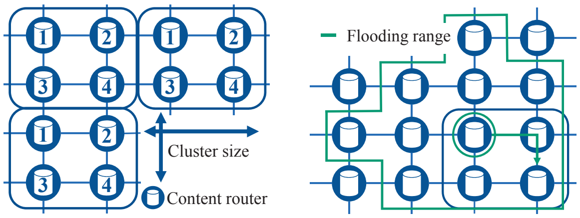
We therefore propose an efficient cluster-based cache distribution scheme to improve the time needed to obtain content as well as cache efficiency in CCN. Our scheme controls a cache distribution range by clustering to cache various chunks near users. We evaluate the effectiveness of the proposed scheme through simulation.

2 PROPOSED SCHEME

Our scheme distributes and obtains cached chunks to/from a limited range of CRs in a domain in order to efficiently obtain various contents from nearby CRs in large scale networks. To distribute and obtain cached chunks over the domain, our scheme extends the previous work proposed in [3]. This scheme uses a hash function that maps chunk identifiers to responsible CRs. When a CR receives assigned chunks, it caches them with priority. To utilize the assigned chunks, CRs that newly cache or discard assigned chunks advertise its information (newly cached/discarded) to all neighbor CRs by flooding techniques to update FIB entries. Through these FIB entry updates, our scheme can obtain distributed chunks from nearby CRs even if they are not on the default path.

ICN '18, September 21–23, 2018, Boston, MA, USA

ACM ISBN 978-1-4503-5959-7/18/09.
https://doi.org/10.1145/3267955.3269012



(a) Clustering (b) Flooding range

Figure 1: Overview of proposed scheme

To control a cache distribution range as well as reduce the time needed to obtain content, our scheme divides all CRs into some clusters with the same size in a domain and assigns hash values of chunks to each CR in clusters beforehand, as shown in Fig. 1(a). Namely, caches of all contents are similarly distributed in each cluster. Note that the cluster size is defined as the number of CRs on a side of square clusters. The advertising range of flooding is limited to the cluster size in order to suppress the overhead of flooding and FIB entry updates as shown in Fig. 1(b). Users thus become effectively obtaining cached chunks from CRs within the cluster size.

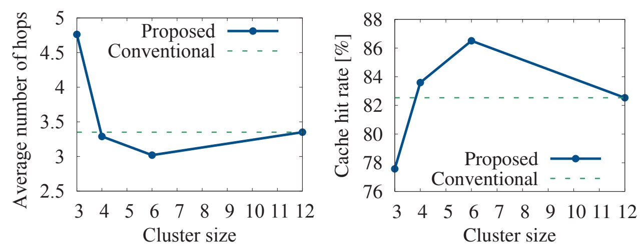
3 SIMULATION EVALUATION

To investigate the efficiency of our scheme, we evaluated it through simulations using Network Simulator ns-3.24.1 after implementation of our scheme. The parameters used in the simulation are summarized in Table 1. We used a grid topology (12×12) having multiple paths, like Fig. 1(a). One server and 12 clients were located on upper and lower sides of the grid, respectively. The cluster size was varied to change the cache distribution range. In this simulation, we assumed that packet losses would not occur so that we could focus on the basic efficiency of our scheme to distribute and obtain cached chunks to/from a limited range of CRs in a domain. The content popularity followed a Zipf distribution [4]. We evaluated the performance of our scheme compared with that of conventional scheme by focusing the average number of hops needed to obtain content and cache hit rates.

Figures 2(a) and 2(b) respectively show the average number of hops needed to obtain content and cache hit rate on CRs within the cluster size from clients when the cluster size varies. From Fig. 2(a), our scheme improves the number of hops when the cluster size is set to a suitable value (4 or 6 in this simulation). This is because our scheme with the cluster size of 4 or 6 achieves higher cache hit rate by mainly obtaining caches from nearby CRs than the conventional scheme as shown in Fig. 2(b). Consequently, our scheme can improve the number of

Table 1: Simulation parameters

Cluster size	3, 4, 6, 12
Bandwidth of each link	1 [GB/s]
Delay time of each link	5 [ms]
Content size	64 [chunk]
Chunk size	1 [KByte]
Cache size on CRs	1000 [chunk]
Cache algorithm	LRU
Number of contents	200
Zipf α	0.8
Simulation time	30 [s]



(a) Average number of hops (b) Cache hit rate

Figure 2: Effect of cluster size

hops needed to obtain content as well as cache hit rate by limiting a cache distribution range with clustering.

4 CONCLUSION

We proposed an efficient cluster-based cache distribution scheme to improve the time needed to obtain content as well as cache efficiency in CCN. Our scheme distributes and obtains cached chunks to/from a limited range of CRs by clustering in order to efficiently obtain various contents from nearby CRs in large scale networks. Simulation evaluations have indicated that our scheme can improve the number of hops needed to obtain content. In our future work, we will evaluate the characteristics of proposed scheme in various environments in detail.

ACKNOWLEDGMENT

This work was supported in part by JSPS KAKENHI Grant Number 16H02806.

REFERENCES

[1] V. Jacobson et al., "Networking named content," *Proc. ACM CoNEXT2009*, pp. 1–12, Dec. 2009.
[2] L. Saino et al., "Hash-routing schemes for information centric networking," *Proc. ACM ICN2013*, pp. 27–32, Aug. 2013.
[3] Y. Sato et al., "Hash-based cache distribution and search schemes in content-centric networking," submitted to *IEICE ComEX*, 2018.
[4] C. Fricker et al., "Impact of traffic mix on caching performance in a content-centric network," *Proc. IEEE INFOCOM NOMEN2012*, pp. 310–315, Mar. 2012.