Probabilistic In-Networking Chunk Marking and Caching for Information-Centric Networks

Saeed Ullah, Choong Seon Hong
Department of Computer Engineering, College of Electronics and Information,
Kyung Hee University, South Korea
Email: {saeed, cshong}@khu.ac.kr

Abstract

In-network caching has remove the compulsion of centralized management entity of overlay caching techniques but in consequences it has generated the problem of storing redundant copies of the same contents in a number of routers. In this paper we propose In-network chunk marking with probabilistic in-network caching that eliminates the caching redundancy for non-popular contents and allows the popular contents to be store at most 2 places in Information Centric Network (ICN). Thus it increase the caching efficiency and capability of content storing as a result it increases the cache hit rate.

1. Introduction and Related Work

Information centric network (ICN) or content centric network (CCN) is realized by Networking Named Content (NNC) model [1] that is designed with the motto of "what" instead of "where". In this new design data is divided into chunks. Each chunk is identified by a unique name. The data must have an originating source where the permanent copy of the data is stored. Whenever an ICN node forward a data chunk it also stores a copy of it in its cache in order to provide it locally for the requests in the near future [1, 2, 5].

ICN nodes are connected by very high capacity lines and it has to forward GBs of data in a second so there a situation may arise that the contents are replaced without any cache hit. To solve this problem the cooperative cache mechanism has been proposed by the research community in the recent fast. There are mainly two types of cooperative caching, overlay caching and in-network caching. In the overlay caching [6] a centralized controlling overlay node is needed that mange and regulate the content cache decisions inside the network. In-network caching is introduced which eliminate the centralized overlay node and the caching decisions are taken by the ICN nodes independently. However this in-network caching faces the problem of cache redundancy.

In the recent fast many researchers have focused on the removal of cache redundancy. Authors in [3] have proposed an idea that probabilistically decide about the content cache. They have tried to minimize the cache redundancy, however their proposal don't ensure the removal of cache redundancy.

In this paper we propose in-network chunk marking in which we remove the problem of cache redundancy. In our proposed chunk marking mechanism at most two copies of the content chunk can be stored inside the ICN. Due to the removal of redundant copies of the same content our proposed scheme will enable the ICN nodes to store a verity of the contents in its cache and as a consequence the cache hit-rate will increase and the network traffic will decrease.

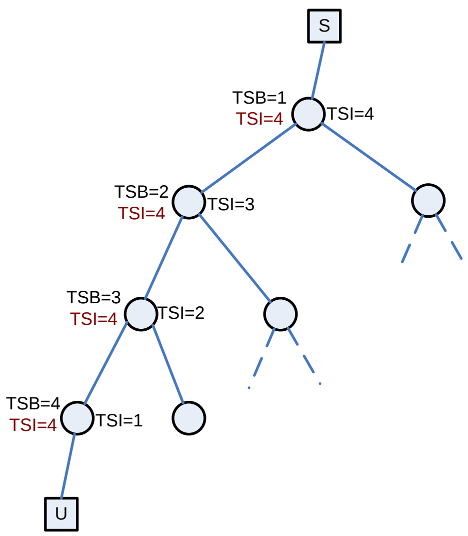

Figure 1: System archetecture.

CS	Meaning
00	Chunk is new in ICN
01	Chunk has been stored once in ICN
10	Chunk has been stored twice in ICN
11	Chunk is stored in cache in the current flow

Table 1: chunk header values and their meanings.

2. System Architecture

For our proposed scheme we use the tree topology of ICN network like [7, 8]. Content store is attached with the root of the tree and the users are attached with the leaves. We propose 3 new header values for the ICN chunk. The ICN request/interest packet contains *Time Since Inception (TSI)* Value while the content chunk contains TSI, *Time Since Birth (TSB)* and *Cache Status (CS)* values. Details of these header fields are discussed in the next section. We use tree structure of the ICN router in which root is the content server and users are attached with the leaf nodes. System architecture of the proposed model is given in Figure 1. We are using the Least Recently Used (LRU) replacement policy for the cache content replacement in our proposed scheme.

3. Proposed Scheme

To eliminate the cache redundancy we need to know the status of the passing content that either any other node in the path has stored it already or not? And secondly how much is it beneficial to store the content in the current node? In the next subsections we make our proposal for these two problems.

3.1. Chunk Marking

The content chunk has to pass through a number of ICN nodes in order to reach from content source to the user. We need 2 bits field in the chunk header to identify the Caching Status (CS) of a passing chunk. The possible status can be, i) the passing chunk has not been stored in any of the ICN node i.e., the chunk has entered the ICN domain for the first time. ii) The chunk that is passing has been taken from the cache of a CCN node and not been changed in the current flow. iii) The chunk has been stored twice in the ICN domain. iv) The chunk is stored in a cache in the current flow. The header field values and their meaning is given in Table 1.

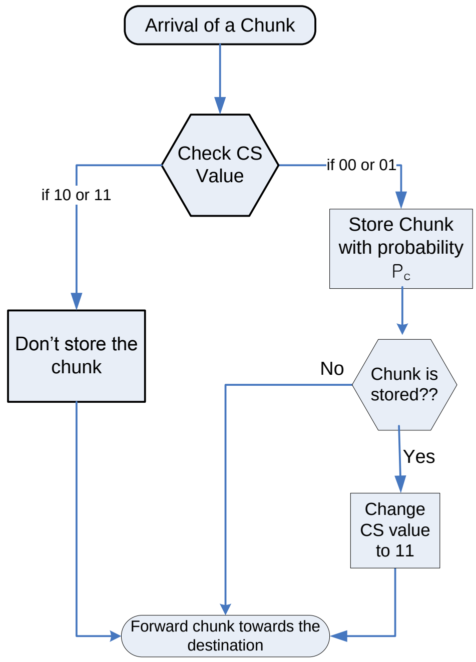

Figure 2: Data chunk storing decision mechanism inside an ICN node.

3.2. Chunk Storing Weightage

For the efficient cache utilization we need to store the chunk in an ICN node that reduce the network traffic and increase the hit rate. To achieve this cache utilization efficiency we need to store the contents in ICN node that is near to the user but since the cache size of ICN nodes is limited therefore we will also need to intelligently use the cache of intermediate router.

Like *Time To Live* (TTL) field in IP packet, our proposed solution will need two header values just like been used in [3], *Time Since Birth* (TSB) and *Time Since Inception* (TSI).

The request message/chunk only contains the TSI value while the response/content chunk contains both TSI and TSB values. In the request message the TSI value is set to 0 by the user and each node on the path towards the content store increment it by 1. Upon receiving the request message the content store fix the TSI value in the data chunk equal to the TSI value in the content chunk and this value remains unchanged during the content chunk delivery, while the TSB value is set to 0 and is incremented by 1 inside each ICN node through which it passes. An example of the TSI and TSB values incremental is shown in Figure 1 in which the Right Side TSI is for the request message while the left side TSI and TSB vales for the content chunk.

Each ICN node take decision of storing the data chunk independently according to the CS, TSI and TSB values. The chunk is store with probability P_c. Probability density function of the P_c is given in equation (1).

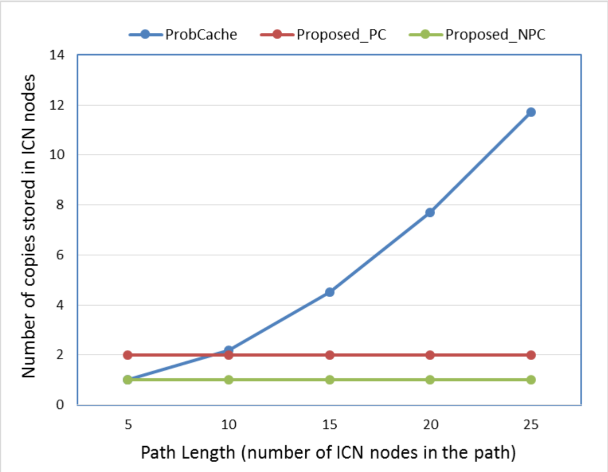

Figure 3: Comparison of our proposed scheme with ProbeCache

$$P_c = \begin{cases} \frac{TSB}{TSI}, & if\ CS = 00 \\ \frac{TSB}{2(TSI)}, & if\ CS = 01 \\ 0, & Otherwise \end{cases} \quad \text{----------------} \quad (1)$$

If the CS value is 11 or 10 then the node will never store the chunk while will store the chunk if the CS value is 00 or 01 with probability P_c. Similarly if the content chunk is not stored by the intermediate nodes it is stored in the last CCN node with probability 1. Detail of the caching decision taking is given in Figure 2.

4. Analysis

We have compared our proposed model with the closest related scheme called ProbeCache that was presented in [3]. We have taken the following parameters for making the analysis. Average cache size of ICN nodes is 10 GB and target time window is 10 seconds. A given path can store target time window worth of traffic. We have done all the calculation in matlab and the results are plotted in Figure 3. In the graph Proposed_NPC means the performance of proposed scheme for non-popular contents. Here those contents are called non-popular that are used only once in the by a user and popular contents are those which have been accessed by the users two or more times. Here one thing to be cleared that the curve for ProbCache is just for one time travel of the contents. If the same content is again requested by some other user and it finds a cache hit the content is again stored in the new path inside the ICN nodes with the ProbCache proposed probability while in our case in one time travel the contents can be stored in only one ICN nodes we term this content as non-popular contents.

5. Conclusion and Future Work

In this paper we proposed a new idea to remove the cache redundancy. Our proposed model only allow 1 copy of the content to be store in ICN in one flow and if the content is popular then at most 2 copies of it can be stored inside the ICN. Our proposed scheme increase the cache capacity of the network by not allowing the redundant copies to be stored and thus increase the cache hit rate.

In the future we aim to further enhance our proposed idea by more analysis and will further evaluate our proposed idea by doing intensive simulations.

6. Acknowledgement

“This paper was supported by NIA (National Information Society Agency).”Dr. CS Hong is the corresponding author.

7. References

[1] Van Jacobson, Diana K. Smetters, James D. Thornton, Michael F. Plass, Nicholas H. Briggs, Rebecca L. Braynard, “Networking Named Content”, CoNext, Rome, Italy, 2009.

[2] L. Dong, H. Liu, Y. Zhang, S. Paul, D. Raychaudhuri, “On the cache-and-forward network architecture”, in: Proceedings of IEEE ICC, pp. 1 -5. 2009.

[3] I. Psaras, W. K. Chai, and G. Pavlou. “Probabilistic in-network caching for information-centric networks.” ACM-SIGCOM workshop on Information-centric networking, ICN '12, pages 55-60, New York, USA, 2012.

[4] S. Arianfar, P. Nikander, and J. Ott. “On content-centric router design and implications”. In ACM ReArch Workshop, volume 9, 2010.

[5] Van Jacobson's talk at PARC Forum: "The Good, Bad, and Ugly of Digital Distribution: A Content-centric Networking Perspective on Evolving Network Architecture", February 2011.

[6] Van Jacobson's tutorial at Future Internet Summer School (FISS09) Short course, 2009.

[7] K Katsaros, G. Xylomenos, G. C. Polyzos: “MultiCache: An overlay architecture for information-centric networking”, ELSEVIER Computer Networks, March 2011.

[8] G. Caro glio, M. Gallo, L. Muscariello, and D. Perino, “Modeling data transfer in content centric networking” 23rd International Tele-traffic Congress (ITC), 2011.