

<http://ansinet.com/itj>

ITJ

ISSN 1812-5638

# INFORMATION TECHNOLOGY JOURNAL

**ANSI***net*

Asian Network for Scientific Information  
308 Lasani Town, Sargodha Road, Faisalabad - Pakistan

## A Cache Strategy in Content-centric Networks Based on Node's Importance

Yuanjun He, Yi Zhu, Ya'nan Ni, Jia Shi and Na Zhu  
School of Computer Science and Telecommunication Engineering, Jiangsu University,  
Zhenjiang, 212013, China

---

**Abstract:** Ubiquitous in-network caching is one of the key aspects of Content-centric Networking (CCN) which has recently received widespread research interest. At present, researchers usually use Leave Copy Everywhere (LCE) strategy as the cache decision strategy by default. Experimental investigation shows that LCE apparently is not a good choice because it does not effectively distinguish the importance of nodes on content delivery path. By learning from the advantages of some existing basic cache decision strategies, a new cache decision strategy which called Content Gradually Tend to Important Node (CGTIN), is proposed in this study. CGTIN is a cache decision strategy who tries its best to push more popular contents to more important nodes and extend the survival time of these more popular contents. In order to test the performance of CGTIN, programs of 4-level binary tree topology are developed by MATLAB. Simulation results show that while compared to LCE, both the SSHP and AHD of CGTIN has an apparent decline while compared to MCD, the SSHP of CGTIN has an apparent degradation and the AHD of CGTIN is exact same when not considering the class of CCN content.

**Key words:** Cache decision strategy, node's importance, content centric networks

---

### INTRODUCTION

Content-centric Networks (CCN) has recently attracted significant attention, with various research initiatives and targeting this emerging. CCN is proposed by Palo Alto Research Center (Jacobson *et al.*, 2009). In fact, cache strategy is a hot research area in CCN. CCN node can cache any content that go through it.

At present, researchers usually use Least Recently Used (LRU) strategy as the cache replacement strategy and Leave Copy Everywhere (LCE) strategy as the cache decision strategy by default. Noticing that LCE is not a very good choice, many researchers have been working on the study of new cache decision strategies. Besides LCE, there are still some basic cache decision strategies. Leave Copy Down (LCD) is a strategy that a new copy of the requested object is cached only at the cache that below the location of the hit on the path to the requesting client. Similar to LCD, Move Copy Down (MCD) is a strategy that moving the copy of the requested object from the cache where the hit occurred to its underlying cache. And Leave Copy Probability (LCP) is a strategy that the copy of the requested object may be cached with a probability  $p$  at intermediate CCN nodes.

Researchers have more contributions on this field. For example, unlike universal caching, some researchers propose a cache strategy which the copy of the requested

object is cached only at the cache having the highest probability for a cache hit on the content delivery path (Chai *et al.*, 2012). A cache scheme named ProbCache has been proposed to reduce caching redundancy. This scheme is based on caching capability of paths and weight-based caching and the copy of the requested object may be cached with a calculated probability value at the nodes of the delivery path (Psaras *et al.*, 2012). An age-based cooperative cache scheme has been proposed to reduce network delay and publisher load, in which the age of a replica is related to the popular and the location of this replica (Ming *et al.*, 2012). To evaluate the hit rates in a two-layer topology, four main types of content, including web, file sharing, user generated content and video on demand have been considered (Fricker *et al.*, 2012).

By learning from the advantages of some existing basic cache decision strategies, a new cache decision strategy which called Content Gradually Tend to Important Nodes (CGTIN), has been proposed in this study. CGTIN is a cache decision strategy who tries its best to push more popular contents to more important nodes and extend the survival time of these more popular contents. In order to test the performance of CGTIN, programs of 4-level binary tree topology have been developed by MATLAB. Simulation results show that

while compared to LCE, both the SSHP and AHD of CGTIN has an apparent degradation; while compared to MCD, the SSHP of CGTIN has an apparent degradation and the AHD of CGTIN is exact same when not considering the class of CCN content.

**CACHE DECISION STRATEGY IN CCN**

**Scenario description:** In this study, CGTIN strategy is suitable for any network topology. In order to facilitate explanation and understanding, a m-level k-ary tree topology is used for analysis (Fig. 1). As shown in Fig. 2, it is a content delivery path selected from the m-level k-ary tree which has been marked by the red dotted line in Fig. 1.

**CGTIN strategy:** Based on the discussion above, it is assumed that the flow rate of all nodes in the network have been pre-calculated by network management system and stored into every user. Now a new cache decision strategy is proposed, specifically as follows: as shown in Fig. 2, a cache hit is recorded for a request finding a matching content along the content delivery path. Otherwise, a cache miss is recorded. (1) Previously, each user obtains the flow rate of all nodes on this content delivery path and computes the importance of each node. (2) When a content client initiates a content delivery, the request message records the importance of all the nodes of the content delivery path. It may be inserted into the request packet header and copied onto the content

messages during the data transmission at the server. (3) If a cache hit occurs on an intermediate node and this node is the most important on content delivery path, this content copy will be marked as recently used and not cached at other nodes. Otherwise, if this node is not the most important on content delivery path, the content copy will be removed from this node and only pushed to the node which is just more important than this node for one level. (4) What's more, if the requiring content is achieved at source server, the content copy will be cached at the least important node. The flow chart of CGTIN is depicted by Fig. 3. However, how to reasonably select these important nodes enable to improve the performance of the CCN is a key problem.

**Importance of node:** Because the importance of node is a key problem in this study, it will be discussed in this section. In a fixed network topology, the flow rate (content request rate) of the node will be changed with the difference of cache strategy. It is agreed that the importance of the node is closely related to its flow rate. Some important nodes may be intentionally looked for in the network and the more popular content can be cached in these nodes as much as possible, the less popular content in less important nodes. If the more popular content have been cached in these important nodes, whose content request rate are high, it will be helpful to improve the cache hit probability, especially in a network topology that each cache capacity is far less than the total number of chunks in source server. It is supposed that the importance of a node is influenced by such factors, including the property of other nodes in the network,

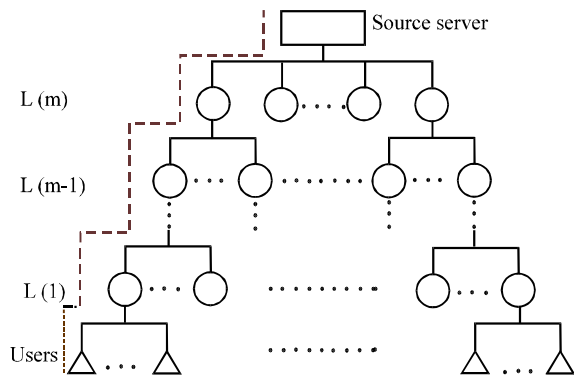


Fig. 1: M-level K-ary tree

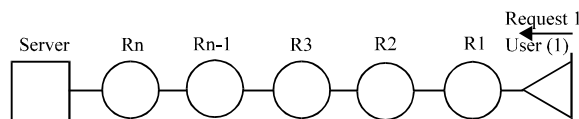


Fig. 2: Content delivery path

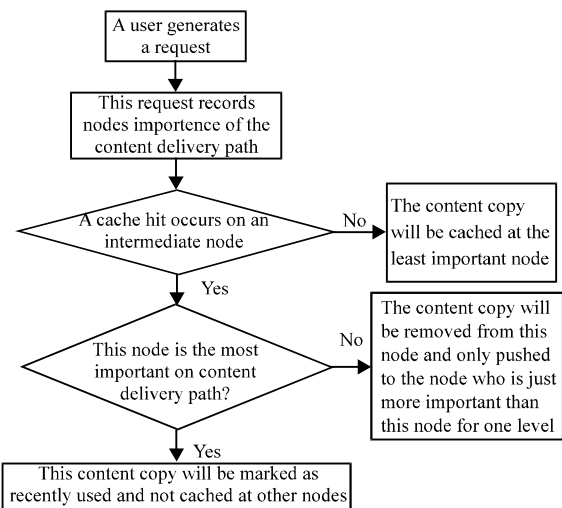


Fig. 3: Flow chart of CGTIN

visiting users' number, the content request rate generated by each user and the distance of this node to each visiting user.

As shown in Fig. 2, a request generated by user (1) will search the content along the content delivery path directed to source server in the network. If the content cannot be acquired at the intermediate nodes, it will be acquired at the source server. Now it is assumed that: (1) Content delivery path has n nodes, Router Ri is i hops from users; (2) The capacity of Router Ri is h(i), the unit is chunk, each chunk have the same size; (3) There are m files in the source server, each file has N(i) chunks, so the total number of chunks in the source server is:

$$\sum_{i=1}^m N(i)$$

(4) At initial stage, each node is full caching by randomly selecting chunks from source server; (5) Zipsf distribution is used to represent the request popularity in the first layer and  $q_k$  represents the request probability of class k content in the first layer, that is  $q_k = c/K^a$ . Here,  $c > 0$  and a represents the concentration of content popularity distribution. The content is divided into K classes by popularity. (6)  $p_i$  represents the hit probability of this request in Router Ri,  $p_i(k)$  represents the cache hit probability of class k content in i layer,  $\lambda_j$  represents the content request rate of user(j) and  $R_i(\lambda_j)$  represents the part of content request rate in Router Ri which is caused by user(j). Based on the above assumptions,  $p_i(k)$  may be got by Eq. 1:

$$p_i(k) = \frac{h(i)}{\sum_{i=1}^m N(i)} \tag{1}$$

Then,  $p_i$  can be calculated by Eq. 2:

$$\begin{aligned} p_i &= \sum_{k=1}^K p_i(k) * q_k \\ &= \frac{h(1)}{\sum_{i=1}^m N(i)} * q_1 + \frac{h(1)}{\sum_{i=1}^m N(i)} * q_2 + \dots + \frac{h(1)}{\sum_{i=1}^m N(i)} * q_k \\ &= \sum_{k=1}^K (q_1 + q_2 + \dots + q_k) * \frac{h(1)}{\sum_{i=1}^m N(i)} = \frac{h(1)}{\sum_{i=1}^m N(i)} \end{aligned} \tag{2}$$

After  $p_i$  is calculated,  $p_2$  will be got by Eq. 3:

$$p_2 = (1 - p_1) * \frac{h(2)}{\sum_{i=1}^m N(i)} \tag{3}$$

While  $p_1$  and  $p_2$  are calculated,  $p_3$  will be got by Eq. 4:

$$p_3 = (1 - p_1 - p_2) * \frac{h(3)}{\sum_{i=1}^m N(i)} \tag{4}$$

And so on,  $p_i$  will be got by Eq. 5:

$$p_i = (1 - p_1 - p_2 - \dots - p_{i-1}) * \frac{h(i)}{\sum_{i=1}^m N(i)} \tag{5}$$

Therefore,  $R_i(\lambda_j)$  will be got by Eq. 6:

$$R_i(\lambda_j) = \lambda_j * (1 - \sum_{s=1}^{i-1} p_s) \tag{6}$$

However, the content request rate of each node in the network is caused by one or more users. If there are l users visiting router Ri in the network, the complete content request rate (flow rate) in router Ri will be got by Eq. 7:

$$\text{flow}(R_i) = \sum_{j=1}^l R_i(\lambda_j) \tag{7}$$

As there are n nodes on content delivery path, the importance of each node may be acquired by Eq. 8:

$$\text{Important}(R_i) = \frac{\text{flow}(R_i)}{\sum_{j=1}^n R_j} \tag{8}$$

### SIMULATION AND PERFORMANCE ANALYSIS

**Simulation tool and performance indicators:** For testing the performance of CGTIN, programs are developed by MATLAB.  $SSHP_k$  (Source Server Hit Probability of class k content) and  $AHD_k$  (Average Hit Distance of class k content) are used as performance indicators which are defined as Eq. 9 and 10. Here  $p_i(k)$  represents the cache hit probability of class k content in i layer when CGTIN is adopted.

$$SSHP_k = 1 - \sum_{q=1}^L p_i(k) \tag{9}$$

$$AHD_k = \sum_{q=1}^L q * p_i(k) + (L + 1) * SSHP_k \tag{10}$$

In order to show the superiority of CGTIN, the existing basic cache decision strategies LCE and MCD are used to compare with it and LRU (Least Recently Used) are uniformly used as the cache replacement strategy. The reasons of selecting LCE and MCD are that: (1) At present

LCE is the mainstream cache decision strategy of CCN. (2) MCD is a decision strategy that when a cache hit occurs at q level, the content copy will be removed from this cache and only pushed to the cache of next hop node which is closer to users. MCD is similar to CGTIN to some extent.

**Simulation results and the analysis of basic indicators:**

Most of simulation parameters come from the study “Impact of traffic mix on caching performance in a content-centric network” (Fricker *et al.*, 2012). A 4-level binary tree of 14 nodes in total is used as the topology; the root node represents the server node and requests are enabled to come from the last levels of the tree and individual users are assumed not connected to backbone network routers; the exponent of the Zipf distribution of content popularity is set to 0.8; the total number of content files of CCN is  $M = 40000$ , divided into  $K = 400$  classes by popularity; each class contains  $m = 100$  content files; each content file is 100 M and splits into chunks of 100 kB; The requests come from each user submit to Poisson distribution with parameter as  $\lambda_i = 40$  content  $\text{sec}^{-1}$ ; the size of cache is set to  $C = 2$  GB at each cache. A total of 1000,000,000 requests are generated to allow enough time for the system to reach a steady state. This simulation costs approximately 53 h. Figure 4 and 5 show the SSHP and AHD of the first ten classes CCN content while the cache decision strategy is LCE, MCD and CGTIN, respectively.

Conclusions can be got from Fig. 4 and 5: (1) Compared to LCE and MCD, the SSHP of CGTIN has an apparent decline and with the increase of content popularity, it decreases even more obviously. That is because the more popular contents will tend to the more important nodes step by step on CGTIN, so that it is more conducive to the realization for the rational allocation of the limited CCN cache resources in network. Actually, the simulation numerical results indicates that when compared to LCE and MCD, the cache hit probability at medial nodes of CGTIN increases 146.8 and 57.3%, respectively. (2) Compared to LCE, the AHD of CGTIN has an apparent decline and with the increase of content popularity, it decreases even more obviously. While compared to MCD, the first class of AHD of CGTIN increases some, the second and third classes is the same and from the fourth class, it decreases. That is because the more popular contents will tend to the more important nodes step by step on CGTIN while the more popular contents will just tend to users step by step on MCD. In this 4-level binary tree topology, the most important nodes are not closest to users, leading to the increasing of the first class of AHD

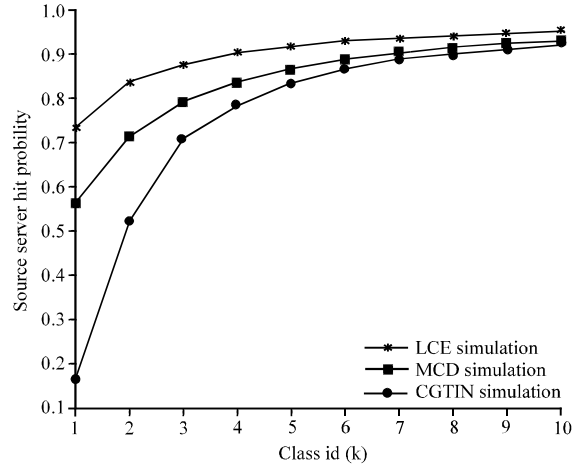


Fig. 4: Source server hit probability comparison of LCE/MCD/CGTIN from the first class to the tenth class CCN content

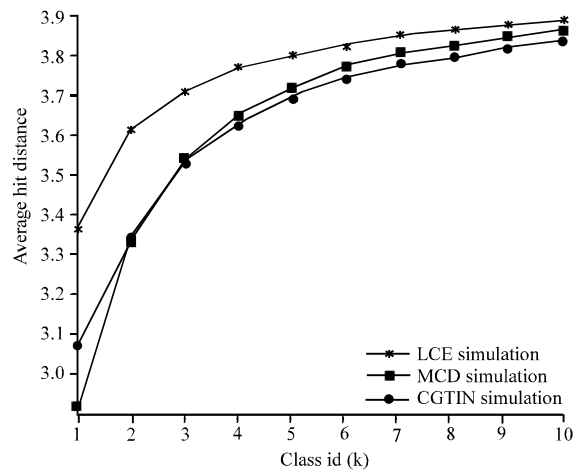


Fig. 5: Source server hit probability comparison of LCE/MCD/CGTIN from the first class to the tenth class CCN content

of CGTIN. Actually, the simulation numerical results shows that, when not considering the class of CCN, the AHD of MCD and CGTIN are exact same.

**CONCLUSION**

Ubiquitous in-network caching is one of the key aspects of content-centric networking which has recently received widespread research interest. The cache decision strategy is one of the core issues of the CCN studying. By learning from the advantages of some exiting basic cache decision strategies, a new cache decision strategy CGTIN is proposed. CGTIN is a cache decision strategy who tries

its best to push more popular contents to more important nodes and extend the survival time of these more popular contents. Then it is tested by developing programs of 4-level binary tree topology on MATLAB. Simulation results show that while compared to LCE, both the SSHP and AHD of CGTIN has an apparent decline while compared to MCD, the SSHP of CGTIN has an apparent decline and the AHD of CGTIN are exact same when not considering the class of CCN.

In this study, a method to measure the importance of nodes on content delivery path is designed and a new cache decision strategy CGTIN is proposed. But the best distribution of important nodes on a sufficiently complex network is not further considered. It is expected that more success on similar directions of CCN research field will be achieved.

#### **ACKNOWLEDGMENT**

The study is financially supported by Modern Education Technology Institute, School of Jiangsu Education Scientific Research with Grant No. 2012-R-22797.

#### **REFERENCES**

- Chai, W.K., D. He, I. Psaras and G. Pavlou, 2012. Cache less for more in information-centric networks. Proceedings of the Part I, 11th International IFIP TC 6 Networking Conference, May 21-25, 2012, Springer-Verlag, Berlin, Heidelberg, pp: 27-40.
- Fricker, C., P. Robert, J. Roberts and N. Sbihi, 2012. Impact of traffic mix on caching performance in a content-centric network. Proceedings of the IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPs), March 25-30, 2012, Orlando, FL. USA., pp: 310-315.
- Jacobson, V., D.K. Smetters, J.D. Thornton, M. Plass, N. Briggs and R.L. Braynard, 2009. Networking named content. Proceedings of the 5th International Conference on Emerging Networking Experiments and Technologies, December 1-4, 2009, ACM, New York, USA., pp: 1-12.
- Ming, Z.X., M.W. Xu and D. Wang, 2012. Age-based cooperative caching in information-centric networks. Proceedings IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPs) March 25-30, 2012, Orlando, FL. USA., pp: 268-273.
- Psaras, I., W.K. Chai and G. Pavlou, 2012. Probabilistic in-network caching for information-centric networks. Proceedings of the 2nd Edition of the ICN Workshop on Information-Centric Networking, August 13-17, 2012, ACM New York, USA., pp: 55-60.