International Journal of Scientific Research in Computer Science, Engineering and Information Technology
© 2017 IJSRCSEIT | Volume 2 | Issue 5 | ISSN : 2456-3307

A Caching Strategy Based on Content Popularity and Router Level for NDN

Anjali
Research Scholar, Panjab University, Chandigarh, India

ABSTRACT

Named Data Networking (NDN) is one of the future Internet architectures and has recently attracted a great deal of attention. NDN adopts named routing and caches all content on routers en-route. It helps to accelerate content distribution and reduces content retrieval latency. However, it also brings a large number of redundant data in the network. In this paper, we propose an in-network caching strategy based on content popularity and router level (CPRL) which caches the content according to its popularity level on the corresponding router to reduce redundant data. The simulation results show that the CPRL protocol can effectively reduce the amount of redundant data in the network and reduce average delay of request, as well as increase the diversity of data while acquiring the similar cache hit ratio compared to the Leave Copies Everywhere (LCE) caching strategy.

Keywords : Named Data Networking; Caching Strategy; Content Popularity; Router Level

I. INTRODUCTION

Internet traffic has grown rapidly in the last few years and most of the traffic could be due to the distribution and retrieval of content. This vigorous growth in traffic brings significant challenge to the current host-centric IP -based network. The users are more interested in content itself rather than the address of the content. To relieve bandwidth pressure caused by the rapid growth of network traffic, Named Data Networking (NDN) [1-2] is proposed as a promising architecture based on contents for future network. The nodes in NDN have the availability of named routing and the in-network caching. NDN uses the routers to cache all the contents that have passed by. It can accelerate content distribution, reduce content retrieval latency, and greatly enhance the utilization of network resources.

In NDN, users send interest package to request the content they needed. Routers receive the interest package and search the matched content in their Content Store (CS). If a matched content is discovered, they will send the data to users. Otherwise, they will forward the interest package to the next routers according the Forwarding Information Base (FIB). Routers can cache the contents which can satisfy the incoming requests instead of forwarding it to the source node. Therefore, NDN can relieve bandwidth pressure and reduce the delay of request.

An important feature of NDN is to manage the in-networking caches with caching strategy. However, the default in-network caching strategy, Leave Copy Everywhere (LCE) [1], caches all the contents in all en-route routers that cause two problems as follows:

The adjacent routers cache the same contents that have caused a large number of redundant data in the network and reduced the diversity of data.

The routers on the path of data replying are treated equally to cache same contents and the importance of each router's location does not be considered. It greatly wastes the storage space of caches on the routers.

In order to solve the above problems, we propose an in-network caching strategy based on content popularity and router level (CPRL) to improve the overall performance of NDN network in this paper. The strategy makes full use of the popularity of the content, and then selectively caches the contents according to their popularity on the matching routers. It improves the utilization of cache resources and reduces the frequency of contents replacement. In addition, on the path of data replying, only one router will be selected to cache the contents. It avoids the same data caching on the adjacent nodes, reduces the amount of redundant data in the network, increases the diversity of data, and improves the cache hit rate.

The reminder of this paper is organized as follows. Section presents the related work. The detailed CPRL strategy is illustrated in section and the simulation environment is shown in Section . Section V shows the performances of our cache strategy. Finally, Section VI concludes the paper.

II. RELATED WORK

In order to solve the problem of data redundancy caused by LCE, that is, all of the contents are cached in all en-route routers, many researchers have proposed different solutions to improve the cache strategy of NDN.

The literature [3] proposes to segment each file and spread them among the routers along the routing path in NDN, putting the number of the former segments close to the customer, and the number of latter segments close to the producer. In literature [4], an efficient popularity-driven cache scheme is developed to dynamically place content replicas on the en-route path in a coordination fashion in NDN, which can effectively reduce the average access hops. The popularity of an object is measured by the request rate for the object. Literature [5] proposes a cache permission policy called Cache-Filter in ICN, which takes the content popularity into account and caches only popular content in routers which is close to users through the collaboration of en-route nodes.

In literature [6], a centrality-based caching algorithm by exploiting the concept of (ego network) betweenness centrality is proposed. By discussing the betweenness centrality of routers, the caching algorithm caches content only on the subset of nodes on the content delivery path. In the case of constrained total storage budget, Literature [7] proposes a cache allocation method based on node centrality to solve the problem of how to allocate memory reasonably on the router. However, literatures [6-7] both cache the content only at the top centrality routers, which will increase the load on that routers and cause high frequency of cache replacement.

Literatures [8-9] take both content popularity and node level into account. Literature [8] proposes a probability caching strategy (MPC) which based on content popularity and matched node level. Meanwhile, the content popularity has been calculated in advance according to the Zipf law. In literature [8], the content popularity is static, which does not take the dynamic changes of the content requests into account in the network. Literature [9] designs a dynamic, self-adaptive method to calculate the content popularity and proposes a novel caching scheme (CRCache) that utilizes a cross-layer design to cache contents in a subset of routers based on the correlation of content popularity and the network topology. However, [8-9] need a lot of extra information or topology of the entire network when calculating the router level, which will increase the storage and computing overhead of routers.

III. CPRL CACHING STRATEGY

The CPRL caching strategy progressively caches the content with high popularity in the node close to the user, in order to improve the cache hit rate and reduce the average request time. This strategy aims to cache the content according to its popularity level on the corresponding node. Adopting the CPRL strategy, request counts of contents with high popularity will continue to increase and the popular content will gradually move to the nodes nearest to users. In this section, we sketch out the environment and assumption under which CPRL is designed. In addition, the details of our proposed CPRL strategy are described. CPRL consists of four main parts, including data structure needed in CPRL, the popularity level of content, the level of node on the path and CPRL caching strategy.

A. Data structure needed in CPRL

In CPRL, we use four table structures: CS, FIB, Pending Interest Table (PIT) and a new table named User Request Table (URT) which is maintained by the first level router. Moreover, we modified interest package and data packet. The details are described as follows.

Modify the interest packet format We added two additional fields to record content popularity level (PLX) item and the number of router's hops (hopsX) en-route in interest packet. Modify the data packet format We added a cache flag (Cl) to record the cache location in data packet. New table: In the first hop router who is next to the user, we store Content Name and User Request Count into URT for calculating content popularity level.

B. The Popularity Level of Content

In network, only a small part of the contents will frequently be accessed by most users, most of the content is rarely requested. That is, users will frequently ask for popular content rather than unpopular content. Caching popular content on nodes close to the user will greatly reduce the user's request latency. In our strategy, the more times the content is accessed, the more popular it is.

In CPRL, the first level routers who are close to the user count the number of requests for requested content, and store content name and request number into URT. The calculation of router level will be mentioned later. We present a simple classification of popular level of content X (PL_X) ccording to the number of requests of content X (n_X) and divide the content popularity into M levels, each of which includes N requests. The detail is described as follows:

$$PL_X = \left\{ \begin{array}{ll} M- \left(\left\lceil \frac{n_X}{N} \right\rceil - 1 \right) & n_X < N*M \\ 1 & n_X \geq N*M \end{array} \right\} \quad (1)$$

Contents whose level is 1 are most popular and those whose level is M are the lowest. When the node replies the matching content to the user, the content will be cached on a router with a corresponding level on the return path according to the level of the content.

C. The Level of Router on path

In CPRL, when the user requests content, the routers on the forwarding path will be assigned different levels according to the number of hops from the user. The routers who are closer to user will be assigned the highest level, and the level of routers that is farther from the user will gradually decrease. In our strategy, we adopt the number of hops from the user to represent the level of routers. The router, which is next to user, is called the first level router in which we will calculate the popularity level of content.

The level of the routers is available only during a certain request and reply period on the path. Namely, other routers in the network will not be assigned router level when they are not en-route. Moreover, each router may have different levels on different forwarding paths. Taking dynamically assignment, our strategy does not depend on the topology of networks and we do not need to cache the information of the routers level and hence decrease the storage pressure on the router.

D. CPRL strategy

This section describes the operation of CPRL strategy in detail and the related pseudo-code is shown in Table 1 and 2.

In CPRL, the consumers send interest packet in which popularity level and hops are both set to 0 to seek the content they needed. After receiving the interest packet, the first router looks for the number of requests in the URT table and calculates the content popularity level according to the Formula 1. Then it adds the content popularity level in interest packet for calculating the caching location and the hops is set to 1. The other routers who receive the interest packet will forward the interest packet after adding 1 to route hops count.

The operation of the interest request phase and the related pseudo-code is shown in Table I.

When providers who own the content that consumer needed receive the interest packet, they will calculate the cache location (Cl). The Cl is denoted as follows:

$$Cl\ \ round\,((\,PL_x - 1) * \frac{hops_X}{M} + 1)$$

Where $hops_X$ indicates the number of hops content X

The routers received the data packet and checked the value of cache flag. If the value of cache flag is 0, they will cache the data packet, otherwise, the value minus 1.

The operation of the data reply and the related pseudo-code are shown in Table II.

TABLE I. THE DESCRIPTIONS OF INTEREST REQUEST PHASE

Pseudo code 1 Interest request phase

```
for each(received interest packet )
    if (find match data in CS) then
        calculate Cl
        send(data)
    else
        if (hops==0) then
            calculate content popularity level
        else
            do nothing
        end if
        hops++
        if (find match content name in PIT) then
            add interface of previous router to PIT
        else
            if (find match content name in FIB) then
                forward interest request according to the FIB
            else
                drop the interest request
            end if
        end if
    end if
end for
```

IV. EXPERIMENT EVALUATION

In this section, the simulation results of CPRL caching strategy are presented, compared and analyzed. We detail the simulation environment and simulation parameters.

A. Simulation Parameter Setting

The simulation configurable parameters are depicted as Table III. In all our simulations, we choose LCE caching strategy as a comparison strategy with our CPRL strategy and record experimental results in 10s, 20s, 30s, 40s, and 50s, respectively. We employ a random cache replacement strategy (RAND) in our simulations. Namely when the cache space is full, RAND will randomly remove a content which is to make room for the new one.

In CPRL, the popularity of contents have been modeled following a Zipf distribution function [10]. In our experiments, we choose the same distribution function to model the content requests. The value of N is 100 and M is 5 when we calculate the level of content.

TABLE II. THE DESCRIPTIONS OF DATA REPLY PHASE

TABLE II. THE DESCRIPTIONS OF DATA REPLY PHASE

Pseudo code 1 Data reply phase

```
for each(received data packet)
    if (find match content name in PIT) then
        if (Cl==1) then
            store content into CS
        else
            Cl --
            forward data packet according to the PIT
        end if
    else
        drop the data packet
    end if
end for
```

TABLE III. SIMULATION PARAMETER SETTING

Parameter	Value
Number of Content	1000
Request Rate	200 req/s
number of content stored per cache	10
Caching Replacement	RAND
Zipf exponent parameter	=1

B. Evaluation Metrics

We use the following metrics to compare CPRL and LCD Caching strategy.

1) Cache Hit Ratio[11]: the probability to obtain a cache hit all along the path from a requester to a cache node;

2) Cached Copies Ratio: the total number of cached replies in CPRL to that in LCE ;

3) Average delay of request: the average response time of the users successfully obtained content which they request.

C. Experiment results and analysis

Cache hit ratio: Figure 1 is a comparison of the cache hit ratio of the CPRL strategy and the LCE strategy. We will discuss the results in Figure 1 from two aspects as follows. On one hand, when the running time is 10s and 20s, the cache hit rate of CPRL strategy is lower than that of LCE strategy. That's because the content popularity level is quite low in CPRL strategy at the initial simulation stage. However, with the increasing of time, the content popularity level is increasing, and the cache hit ratio of CPRL strategy is higher than that of the LCE strategy on the other hand. This shows that CPRL plays a very good role in improving the cache hit ratio of NDN, which satisfies our original intention of designing this strategy.

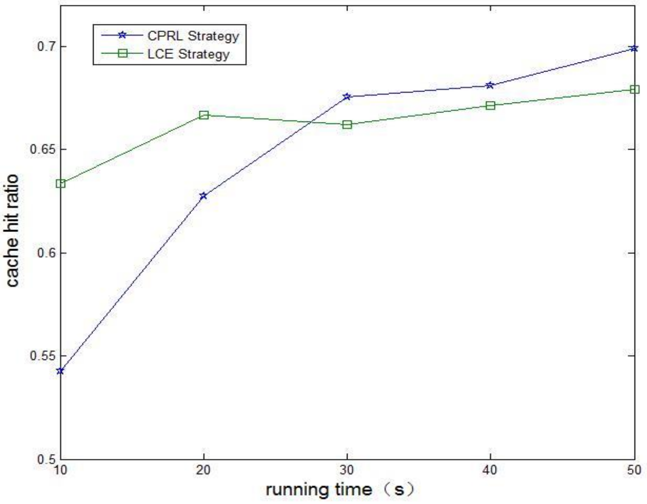

Figure 1. Cache hit ratio

2) Cached copies ratio: Figure 2 is a comparison of the ratio of cached copies in NDN network between the CPRL and the LCE cache strategy. The results in Figure2 show that the ratio of cached copies is always less than one. Namely, the number of cached copies of the CPRL strategy is always less than that of the LCE strategy. So the cache space can be effectively utilized to reduce the content copies in the network and the time overhead of cache. Because CPRL only caches content in one router along the path from a cache node to a user, while LCE strategy caches content in all routers along the path. As illustrated by Figure 2, our caching strategy effectively solves the problem of highly cache redundancy.

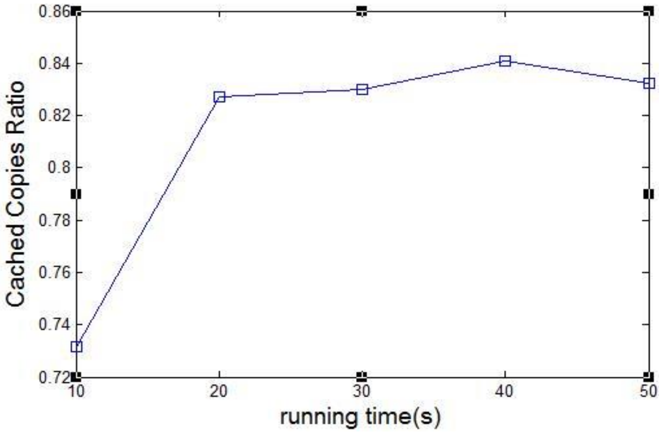

Figure 2. Cached copies ratio

3) Average response time: Figure 3 is a comparison of average response time in NDN network between the CPRL and the LCE cache strategy. The results in Figure 3 show that the average response time of the CPRL strategy is not always higher than that of the LCE strategy. Compared to LCE, the CPRL has a good advantage in terms of average response time and it reduces the average response time by about 25% to 50%. Because CPRL caches popular content in routers who is nearest to the user, while LCE strategy caches all contents in nearest routers where the popular contents may be frequently replaced. As illustrated by Figure 3, our caching strategy effectively improves the average response time of users.

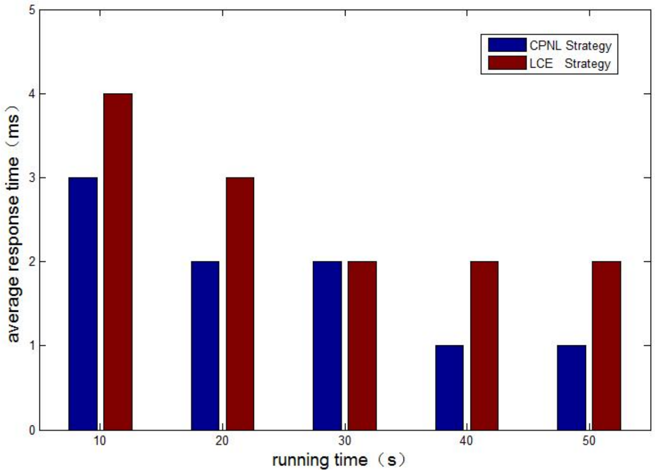

Figure 3. Average response time

V. CONCLUSION

In this paper, we proposed the CPRL caching strategy for NDN. The strategy makes full use of the popularity of the

VI. ACKNOWLEDGMENT

This research is supported by the National Nature Science Foundation of China, NSFC (Grant No. 61363079) and the Research Project of Higher Education School of Inner Mongolia Autonomous Region under Grant NJZY16020.

VII. REFERENCES

[1]. Anjali, FUTURE INTERNET ARCHITECTURE AND INFORMATION CENTRIC NETWORKING: REVIEW , Proceedings of the 11th INDIACom; INDIACom-2017; IEEE Conference ID: 40353 2017 4th International Conference on “Computing for Sustainable Global Development”, 01st - 03rd March, 2017 Bharati Vidyapeeth's Institute of Computer Applications and Management(BVICAM), New Delhi (INDIA).L.

[2]. Anjali Goyal, Anu Gupta and Ajay Guleria. “NAMED DATA NETWORKING: AN EMERGING INTERNET PARADIGM”, Panjab University, Research Journal.,vol. 65, 2015, pp. 41-48

[3]. Rezazad, Mostafa, and Y. C. Tay. "CCndnS: A strategy for spreading content and decoupling NDN caches." Ifip NETWORKING ConferenceIEEE, 2015:1-9.

[4]. Li, Jun, et al. "Popularity-driven coordinated caching in named data networking." Eighth Acm/ieee Symposium on Architectures for Networking & Communications Systems IEEE, 2017:15-26.

[5]. Feng, Bohao, et al. "Cache-Filter : A Cache Permission Policy for Information-Centric Networking." Ksii Transactions on Internet & Information Systems 9.12(2015):4912-4933.

[6]. Wei Koong Chai, Diliang He, Ioannis Psaras, and George Pavlou.

[7]. _&DFKH_³/HVV_IRU_0RUH´_LQ_,QIRUPDW LRQ-Centric Networks." International Ifip Tc 6 Conference on NETWORKINGSpringer-Verlag, 2012:27-40.

[8]. Wang, Yonggong, et al. "Design and Evaluation of the Optimal Cache Allocation for Content-Centric Networking." IEEE Transactions on Computers 65.1(2015):95-107.

[9]. Yao Li, Tiankui Zhang, Xiaogeng Xu, Zhimin Zeng, Yinlong Liu, " Content popularity and node level matched based probability caching for content centric networks," in IEEE/CIC International Conference on Communications in China, 2016:1-6.

[10]. Wang, Wei, et al. "CRCache: Exploiting the correlation between content popularity and network topology information for ICN caching." ICC 2014 - 2014 IEEE International Conference on Communications IEEE, 2014:3191-3196.

[11]. L. Breslau, P. Cao, et al. "Web caching and Zipf-like distributions: Evidence and implications. "In Proc. INFOCOM, pages 126±134, 1999

[12]. Bernardini, C, T. Silverston, and O. Festor. "MPC: Popularity-based caching strategy for content centric networks." IEEE International Conference on Communications IEEE, 2013:3619-3623.