International Journal of Future Generation Communication and Networking
Vol. 9, No. 4 (2016), pp. 285-300
http://dx.doi.org/10.14257/ijfgcn.2016.9.4.26

A Hybrid Coordination Approach of In-network Caching for Named Data Networking

Chih Yen Chang[1] and Ming Sang Chang[2*i]

[1]*Graduate Institute of Communication Engineering, National Taiwan University, Taiwan.*

[2]*Department of Information Management, Central Police University, Taiwan.*
E-mail [1]gsmmcc@gmail.com [2]mschang@mail.cpu.edu.tw

Abstract

In-network caching is an important feature of Information Centric Networking (ICN). All routers in ICN will maintain caching ability, and, for each request, contents are cached at routers along the delivery path. This kind of on-path caching can reduce overall retrieval time. However, the same contents may end up to be cached all over the network causing wastes of caching space and decrease content diversity. Coordinated caching policy decides which contents to be stored at which routers cooperatively for better utilizing cache space. Nevertheless, with limited storage, most of coordinated caching policies use all the caching capacity to store popular contents. It may raise a fairness issue in which unpopular contents can not access caching space and thus lower overall performance. In this paper we propose a hybrid coordinated caching algorithm. By splitting cache space into two parts running different mechanisms respectively, it provides accessibility for both popular and unpopular contents. The evaluation results show that fairness is uplifted while maintaining high hit ratio and minimized latency.

Keywords: *Content Centric Networking, Named Data Networking, In-network Caching*

1. Introduction

The Internet Protocol (IP) traffic has risen exponentially because of the continuous increasing of Internet penetration degree. According to Cisco's report [1], it claims that the number of devices connected to IP network will be three times of global population in 2019, up from nearly two times in 2014, which generate two zettabytes per year of global IP traffic. In particular, the video traffic will represent 80 percent of overall IP traffic by 2019, up from 64 percent in 2014. These video traffics come from TV, video on demand, Internet and peer-to-peer sharing. Companies such as Netflix and BBC provide services that user can access on-line video on demand with available high bandwidth. Websites offering user-generated video like YouTube also take a large amount portion of video traffic. The above forecasts and evidences show how fast the Internet demand grows and how far the usage of Internet is away from its original purpose.

The Internet architecture was originally designed and focuses on communicating entities. However, the mainstream of Internet usage has been shifted now. People use Internet to fetch contents, like audio and video, much more often than just communicating with remote entities. As using location-centric model for underlying routing paradigm on Internet, datagrams are transferred from one routable endpoint to another based on location information. This kind of model restricts how contents to be disseminated and hence affects the performance of content distribution. In addition, when requesting, clients usually care more about what the content is than where it comes from.

[i] Corresponding author: Dr. Ming Sang Chang, Department of Information Management, Central Police University, Taiwan. Email address: mschang@mail.cpu.edu.tw

ISSN: 2233-7857 IJFGCN

A new network paradigm, Information Centric Networking (ICN), has been proposed to facilitate content distribution. ICN adopts host-to-content architecture instead of host-to-host one, which is utilized by IP network. By mapping each content with a unique name, clients can fetch desired contents without knowing the location of sources. Routers in ICN also perform traffic directing functions based on content names. There are some papers that realize concepts of ICN in slightly different approaches. Taking Named Data Networking (NDN) [2] as an example, when requesting each of the contents, the consumer will send a packet containing the name of specific content. While receiving the packet, routers will extract the name in packet header, look up the tables maintained that record routing information and forward the packet to specific out-port. After receiving the packet, host who owns the content will return content packets back to the requester. This kind of behavior is also called Pull Model, i.e., content acquisition needs to be triggered by consumers instead of content producer. Due to lack of location information, content producers will only know that someone has requested the content but not able to find out whom the specific one requests it, and vice versa, consumers will receive data without knowing where the content producer is. It then gives certain degree of network privacy, which cannot be achieved by Internet architecture.

Caching is also an important feature for benefiting content delivery. In IP architecture, content providers frequently employ content delivery network (CDN) to offer end-users a better quality of transferring contents. When requesting specific contents, CDN will re-direct the request to a server which is comparably closer to end-users instead of the original content producer. It reduces the retrieval time and improves overall performance. Nevertheless, CDN needs to be operated on top of network layer in order to fit IP architecture and will cause heavy overheads while performing. Besides, maintaining CDN is an expensive and hard task for content providers. For ICN, each router maintains caching ability, which is called in-network caching. When serving a request, the data packet will be cached at every router along the path from the content producer to the consumer. This on-path caching behavior has several advantages, such as -

- Multicast support - In IP network, bandwidth is heavily wasted to perform multicast. Nonetheless, things are changed in ICN. With caching ability of routers, contents can be disseminated by edge routers instead of original content producers that saves large amount of resources in network.
- Packet retransmission - When traversing through unstable channels, transmission usually experiences packet losses. For reliable transport protocols, the request will be retransmitted to make sure the transmission completeness. It would be more efficient to retransmit contents from routers closer to end-users than from origin servers in ICN.

Although on-path caching gives benefits on content distribution, it still has some side effects. As data being cached at every router along the delivery path, by chance, the same replicas may be stored all over the network, especially for contents with higher popularity. From a global point of view, it wastes caching space by keeping the same contents at multiple routers. As a consequence, the global hit ratio will be reduced due to cache miss of unpopular contents and thus lowers the overall network performance. Due to the insufficiency of on-path caching, different caching schemes have been proposed to improve the performance of in-network caching in ICN [7] [11].

There are two types of caching scheme in ICN, non-coordinated and coordinated. For non-coordinated caching, routers maintain their own policies and work independently. They all run the canonical caching policy respectively which is usually based on historical usage or requesting frequency. For coordinated caching, routers work jointly to store different contents. By means of information exchanged between routers, the coordinated scheme can make caching decision based on specific performance objectives. W. K. Chai et al. [3] proposed a non-coordinated caching algorithm that place contents into specific routers chosen via betweenness centrality that is measured by the number of times a node

being passed. For a node that lies in cross point of many content delivery paths, it's more likely for users to get cache hits that represents a higher betweenness centrality. Caching content at nodes only with high betweenness centrality can save caching space and enhance hit ratio. Chadi Barakat et al. [4] proposed an off-path caching policy by allocating contents cooperatively to pre-defined routers. Contents with higher popularity would be cached at routers closer to end-users so as to minimize the average latency. S. Guo et al. [5] suggested more popular contents should be cached in routers with lower costs. The cost denotes average cost of accessing specific content that can be measured by routing weights or number of requests received.

Unable to make caching decision from global information, non-coordinated caching can hardly exploit caching space well. In spite of saving the communication cost, the caching space is ether wasted or not fully used with non-coordinated policy. In contrary, coordinated caching performs relatively better in utilizing caching space. However, the common policies of coordinated caching usually store only popular contents. Since caching capacity is limited, the unpopular contents will not be maintained in such conditions and might cause the reduction of global hit ratio. It also raises the fairness issue: does each content have the equal right to benefit from caching capacity? Besides, the more information required to exchange between routers would generate more traffic overheads inside the network. Computation for making caching decision is sometimes costly that makes policy impractical to deploy.

To overcome these problems, we proposed a cooperative caching algorithm which also leverages caching capacity for on-path caching. It then splits each router's storage into two parts, for running coordinated and non-coordinated scheme respectively. By using this hybrid caching policy, we claim that it can avoid drawbacks and take advantages from both types of caching. For determining the capacity split ratio, Yanhua Li et al. [6] gave an analysis by considering the trade-off between communication cost and routing performance. We consult their model and apply the concept in our study with modifications to decide the partitioning fraction. In our method, for non-coordinated part, routers follow on-path caching scheme to cache contents which is simple and widely used in ICN approaches. Most of all, on-path caching would allow less popular contents to facilitate router storage. For coordinated part, each of routers keeps different top-ranked popular contents respectively to avoid caching redundancy. Moreover, the placement of these top-ranked popular contents is based on the goal to minimize latency perceived by end-users. Our proposed algorithm has following advantages –

- Place contents in chosen routers that minimizes average retrieval time of a request
- Caching space for non-coordinated usage can be utilized by unpopular contents, and hence increases global hit ratio and fairness
- Balance in-network traffic by distributive storing popular contents among routers
- No need of exchanging information on-line that eliminates the communication overheads

The rest of this paper is organized as follows. In section 2, we show the literature survey of existing related works for reference. The system model of our proposal is presented in section 3. In section 4, it gives the simulation results and evaluations of our proposed algorithm. Finally, section 5 concludes the study we have done.

2. Related Works

Cache decision policy determines how to place contents at cache nodes. It can be generally classified into two approaches, non-coordinated and coordinated. Following will introduce relevant works using these approaches respectively [11][18][21].

2.1. Non-coordinated Caching Policy

Most of ICN approaches use the simplest caching policy, i.e., Leave Copy Everywhere (LCE) or on-path caching, like CCN/NDN [2], DONA [8], PSIRP [9] and NetInf [10]. Each router along the path between data source and the consumer will maintain a copy of data. Copy with Probability [12] claims to cache the requested data at each node with a given probability p along the path. With the policy, the cache redundancy can be decreased. It then degenerates to LCE when probability p = 1.

2.2. Coordinated Caching Policy

Cache coordination can further be classified as either implicit or explicit according to the degree of autonomy in making cache decision. For implicit coordination, each node needs not to know state information of other nodes or only requires minor information. Nodes have ability to autonomously determine whether to cache objects or not, but still co-working with cache system. On the other hand, in typical explicit coordination schemes, the calculations of placing each object are done by using cache network topology, object access pattern and each cache's state. These kinds of prerequisite information are either obtained by online or offline communication.

2.2.1. Implicit Coordination: Leave Copy Down (LCD) [12][13] and Move Copy Down (MCD) [14] are two similar implicit approaches. In LCD, when a cache hit occurred, the object is only cached at the router closest to the requester that avoids large number of identical copies. MCD considers copy redundancy more aggressively. The copy would be only kept at the router closet to requester while the object at the original hit node will be dropped. These two approaches pull objects down to the edge network in common. Another approach takes concept of probability into coordination. For Randomly Copy One (RCOne) [15], the object copy will only be cached at single router selected randomly along the content return path. Probabilistic Cache (ProbCache) [16] sets probability to cache objects in each node with inversely proportional to distance from the requester to data source. Thus, nodes closer to the requester have higher probability to cache the object copy. It pulls the object to network edge as well as reduces the copy redundancy. The above approaches are concluded in Figure 1.

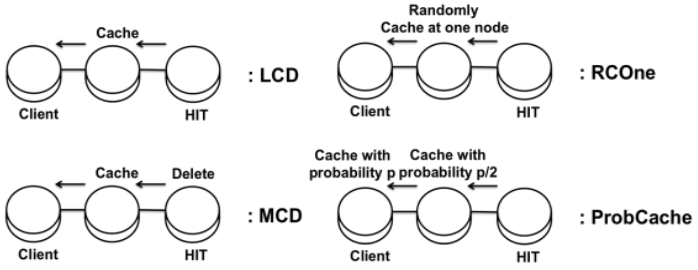

Figure 1. Implicit Coordination Approaches

2.2.2. Explicit Coordination: Nodes with explicit coordination can vary in scope. Usually, there are three types of approaches with varying scopes: path coordination, neighborhood coordination and global coordination. Path coordination means that cache nodes involved in coordination are only those along the path between requesting client and content owner. This approach is typically achieved by inserting the information needed for coordination into content request packet. For instance, the information may be state of each cache node or object requested frequency at each cache node. When request received, the node with a hit will use the information to compute the optimal content placement policy. S. H. Lim et al. [17] proposed Inter-chunk Popularity-based Edge-first Caching (IPEC). IPEC places popular contents at edge cache nodes with a pipeline fashion. Cache pipeline means

to place exclusive content chunk sequentially along the path from client to server. It first divides content into several blocks, and the most popular chunks will be put into the edge router nearest to end-user with following chunks gradually placed along other routers in path. W. Chai et al. [3] proposed a coordination scheme considering multiple paths information. Cache decision is made relying on Betweenness Centrality (BC), which measures number of times a single node lies on transmission path between all pair of nodes in a network topology. Betweenness Centrality is defined in (1) that $\sigma_{i,j}$ is the number of content delivery paths from i to j, and $\sigma_{i,j}(k)$ is the number of content delivery paths from i to j passing through node k. V denotes the set of all nodes in network. The higher BC value a node has, the more likely it is to get a cache hit. During forwarding process, each router along the path inserts its BC value into the header of the requesting packet and the hit node decides to place copy only at node with the highest value.

$$\text{Betweenness Centrality, } C_B(k) = \sum_{i \neq j \neq k \in V} \frac{\sigma_{i,j}(k)}{\sigma_{i,j}} \quad (1)$$

Neighborhood coordination means that coordination tasks are done together with a node's neighbors. The definition of neighbors can vary in scale. For example, it may include a node's direct neighbors or two-hop neighbors. E.J. Rosensweig et al. [19] proposed a scheme that cache node only holds the copy when all other nodes in neighborhood do not own it. Z. Li and G. Simon [20] proposed an approach that group of neighbor nodes decide caching decision by hash function. When an object arrived, the hash function determines which node in the neighborhood to cache the copy. It then avoids the same copy from over-duplicated in cache system.

Global coordination leverages all of nodes in the network together. Typically, cache nodes information, object access frequency per node and distance between nodes are treated as priori knowledge in order to perform the optimal object placement. The goal of placement policy may differ, such as to minimize requesting latency or to balance network traffic. S. Guo et al. [5] makes placement decision by content popularity ranking. The popularity ranking is generated by exchanging information between routers. They proposed a self-adaptive algorithm to solve the inconsistency of content popularity ranking among cache nodes. Object with higher popularity rank would be cached at router with lower accessing cost, which is defined in (2). I_i denotes the average number of interests received by router i and d_{ij} is the link cost of node i and j. V denotes the set of all nodes in network.

$$\text{Cost}_i = \frac{\sum_{i \neq k \in V} I_k d_{ki}}{\sum_{i \neq k \in V} I_k} \quad (2)$$

C. Barakat et al. [4] proposed a content placement scheme that the decision aims to minimize requesting latency based on object access frequency of each node and path length from each node to edge router. The target formulation is shown as (3). $A_{r,c}$ denotes the element of allocation matrix A. If content r is placed at router r, $A_{r,c} = 1$. Otherwise, $A_{r,c} = 0$. C is the set of all contents and R is set of all cache nodes in network excluding edge nodes. E is set edge nodes directly connecting to clients. $\tau_{c,e}$ denotes the demand for content c seen at edge router e. $d_{e,r}$ is the shortest path length between e and r. It results in that content with higher average access frequency would be placed at nodes with shorter path length to edge nodes. Besides, only popular contents can be cached in the network and each content would be kept in single node so as to reduces copy redundancy.

$$\sum_{c\in C}\sum_{e\in E}\tau_{c,e}\sum_{r\in R}A_{r,c}d_{e,r} \quad (3)$$

2.3. Observation

Non-coordinated caching provides the simplest but least optimal solution for making cache decision. The copy redundancy is inevitable and it lacks ability to achieve more complex performance goals. Implicit coordination and path coordination give sub-optimal solution with none or little information exchanging needed. It hence lowers the decision complexity and communication overhead. Global and neighborhood coordination are most complicated ones but providing optimal solution for content placement. Different performance goals can be implemented via these approaches. However, the heavy communication overhead as in [5] and computation with high complexity [4] may raise the difficulty of deployment. Not to mention that there are still lots of issues can be improved, such as cache fairness and biased utilization of links, these factors all make large-scale coordination more difficult to design.

3. System Model and methodology

As mentioned in previous section, most of global coordination approaches make cache decision based on content popularity and the specific goals they want to achieve such as minimizing latency or balancing network traffic. The objects are ranked by popularity based on either information exchanged by routers or probability distribution. With limit global cache capacity, only selected contents can be stored in cache space. Moreover, in order to increase content diversity, each router would cache contents mutually exclusive, i.e., the same copies will not exist simultaneously in multiple cache nodes. Usually, the contents with higher ranked popularity would have higher priorities of placement in order to increase the amount of requests that can be served by caching system. We claim that only store popular content is an aggressive way for enhancing caching performance and can be further discussed.

Consider the following case: There are 1000 different contents that could be requested and routers in network can store 100 entities cooperatively. For the approaches mentioned before, they would cache contents with popularity ranked from 1 to 100. It means that the 101st to 1000th content requests need to be served by the origin server. These contents, which take account of 90 percent of all kinds, may consume more resource and decrease the overall performance. We propose a way to leverage caching space for the use of on-path caching. With on-path caching, contents, either popular or unpopular ones, have ability to be cached in the caching system. We give the right to contents ranked after 100th for being cached and hence increases the average hit ratio. The main concern now is how to determine the fraction of cache space used for non-coordinated caching scheme, which in our study is on-path caching, to improve network performance?

Yanhua et al. [6] have derived a model and give a solid analysis to determine how to divide caching capacity for conducting coordinated caching scheme in NDN. The decision is mainly based on tradeoff between cost of exchanging information among routers and the network performance. We leverage their model with modification in our study.

3.1 System Model

In the NDN environment, we consider the network of a single administrative domain. As Figure 2 shows, our network model includes three sectors, composed of origin server, routers and clients. Server is the abstraction of data source that can serve all the requests.

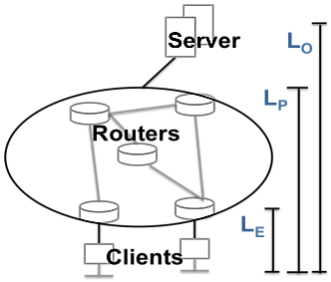

Figure 2. Network System Model

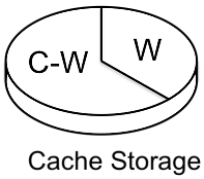

Figure 3. Abstraction of Storage Division

All the routers in the network can perform both routing and caching. There are two types of routers, peer router and edge router. Edge routers are nodes directly connected to clients on network edge. Peer routers are nodes inside the network that do not connect with end-users. Denote R to be the set of routers in network and R^* to be the set of edge routers that $R^* \subset R$. There are n routers in the network including peer and edge routers. Each router maintains the same caching capacity that can cache C entities. Assume each of contents is chunked into identical size that normalized to be one. It means that, for each router, it can store C contents, i.e., capacity of each cache node is C where $C > 0$. We denote the storage used for coordinated caching in each router is W that $W \in [0, C]$ as Figure 3. In each router, the capacity used for non-coordinated caching will be $C - W$. With n routers in network, they can jointly store $n \cdot W$ contents in total. In order to minimize copy redundancy, routers only keep contents that are exclusive for each other.

Content popularity has been proved to follow Zipf's distribution in many studies [22] [23]. The Zipf's distribution is showed in (4) where s denotes a parameter of the exponent characterizing the distribution. N denotes the number of elements that, in our model, represents number of content types where $N \gg 1$. $f(i; s, N)$ is likelihood of the i-th ranked element. $H_{N,s} = \sum_{j=1}^{N} j^{-s}$ is the N-th generalized harmonic number of order s. $F(k; s, N)$ denotes cumulative probability of elements ranked from 1 to k as (5).

$$f(i; s, N) = \frac{1/i^s}{\sum_{j=1}^{N}(1/j^s)} = \frac{1/i^s}{H_{N,s}} \quad i = 1,2, \ldots \tag{4}$$

$$F(k; s, N) = \sum_{i=1}^{k} f(i; s, N) = \frac{H_{k,s}}{H_{N,s}} \quad k = 1,2, \ldots \tag{5}$$

Latency is an important index for network performance. The latency here can be further clarified as shown in Figure 2. In the client's perspective, we denote L_E as the average latency of serving requests from edge routers. L_P is the average latency of obtaining an object from a peer router. L_O denotes the average latency of serving contents from origin servers. When a request cannot be served by edge routers, it will then try to fetch the content from peer routers. If still not finding, the request would be served by the origin server. The relation can be straightforward that $L_E < L_P \leq L_O$. The notations are concluded in Table 1.

Table 1. Notation

Notation	Meaning
C	Cache capacity at each router in the network with C>0
W	Capacity used for coordination at each router W∈[0,C]
n	Number of routers in the network
L_E	Average latency from client to edge routers
L_P	Average latency from client to peer routers

L_O	Average latency from client to the origin server
F(k,s,N)	Cumulative frequency of content requests ranked from 1 to k N for total number of contents that N>>1 S for Zipf parameter

3.2 Methodology

Our goal is to minimize average retrieval latency while maintaining hit ratio. Thus, contents will higher popularity will be placed in coordinated part of cache space at first. The placement manner is that contents with higher popularity will be put at routers by descending order of average latency between edge nodes, i.e., more popular contents need to store at routers with lower latency, in order to minimize average retrieval delay.

For storage used for running on-path caching, it will be occupied by popular contents most of time. Therefore, for each router, we assume that it will cache contents with popularity ranked from 1 to $C - W$. The rest capacity for coordination, with n routers inside the network in total, can jointly store $n \cdot W$ contents. For the sake of reducing copy redundancy, each of contents would only be placed at only one router in network. In the coordination manner, contents with popularity lower than those cached in non-coordinated, ranked from $C - W + 1$ to $C - W + nW$, are stored individually among routers in the network. Hence, the average latency to satisfy a request, $L(W, L_E, L_P, L_O)$, can be derived in (6). The coordination fraction W is then chosen to minimize the $L(W, L_E, L_P, L_O)$.

$$L(W, L_E, L_P, L_O) = F(C - W, s, N) \cdot L_E + [F(C - W + nW, s, N) - F(C - W, s, N)] \cdot L_P + [1 - F(C - W + nW, s, N)] \cdot L_O \quad (6)$$

Currently, the average latency is estimated by only average latencies between clients to hit nodes, i.e., routers or server. For being more precise, the placement of contents stored cooperatively can be further discussed. Let l_{r,r^*} be latency of shortest path between edge router r^* and router r that $r^* \in R^*$ and $r \in R$. For $r = r^*$, $l_{r,r^*} = 0$. L_r denotes the average latency from node r to other edge nodes as shown in (7). Z_k is content with popularity ranked k, where $k = 1,2, \dots N$. Z_{gi} denotes i-th group containing W contents ranked from $C + (i - 2)W + 1$ to $C + (i - 1)W$ where i is the ranking of group popularity and $i = 1,2, \dots n$ since only n groups of contents can be cached inside the network. $F_{Z_{gi}}$ represents the cumulative frequency of content group i as (8).

$$L_r = \frac{1}{n} \sum_{r^* \in R^*} l_{r,r^*}, \forall r \in R \quad (7)$$

$$F_{Z_{gi}} = \sum_{j=C+(i-2)W+1}^{C+(i-1)W} f(j, s, N), \quad i = 1,2, \dots n \quad (8)$$

The cache decision can be expressed via Allocation Matrix A with size $= n * n$. With value being Boolean type, $A_{r,Z_{gi}}$ represents the element of matrix A. If content group Z_{gi} is placed at router r, $A_{r,Z_{gi}} = 1$. Otherwise, $A_{r,Z_{gi}} = 0$. The goal of placement in coordinated manner is to compute the Allocation Matrix A so as to minimize the average retrieval latency (9).

$$\min \sum_{i=1}^{n} \sum_{r=1}^{n} A_{r,Z_{gi}} \cdot F_{Z_{gi}} \cdot L_r \quad (9)$$

The average retrieval latency of contents stored cooperatively contained in (9) can then be merged into (6) by replacing the second term. The average latency to satisfy a request, i.e., $L(W, L_E, L_P, L_O)$, hence becomes (10)

$$L(W, L_E, L_P, L_O) = F(C - W, s, N) \cdot L_E + \sum_{i=1}^{n} \sum_{r=1}^{n} A_{r,Z_{gi}} \cdot F_{Z_{gi}} \cdot L_r + [1 - F(C - W + nW, s, N)] \cdot L_O \quad (10)$$

Finally, the optimal coordination fraction W^* in each router and optimal Allocation Matrix A^* can be obtained simultaneously by making average latency minimized as (11).

$$(W^*, A^*) = \text{argmin}_{(W, A_{r,Z_{gi}})} \{ F(C - W, s, N) + \sum_{i=1}^{n} \sum_{r=1}^{n} A_{r,Z_{gi}} \cdot F_{Z_{gi}} \cdot L_r + [1 - F(C - W + nW, s, N)] \cdot L_O \} \quad (11)$$

The constraints are shown below. First, the value of elements in Allocation Matrix A can only be either 1 or 0 which represents if content group Z_{gi} is cached at router r or not (12). The cache space only stores as many popular contents as it can handle until the space is full (13). Each group of contents will be cached in only one router (14), with each router can only accept one content group (15).

$$A_{r,Z_{gi}} \in \{0,1\}, r \in R \quad (12)$$

$$A_{r,Z_{gi}} = 0, \forall r \in R, i = 1,2 \dots n \quad (13)$$

$$\sum_{r \in R} A_{r,Z_{gi}} = 1, i = 1,2 \dots n \quad (14)$$

$$\sum_{i=1}^{n} A_{r,Z_{gi}} = 1, \forall r \in R \quad (15)$$

4. Simulation Results

In this section, we will show the simulation environment and the simulation results in comparison to other approaches.

4.1 Environment

We adopt NS-3 network simulator to simulate our proposed method. The simulation is done on top of Abilene topology. Abilene is an educational backbone network established by Internet 2 group [24] and is also widely used for evaluating cache performance [5][6]. The topology consists of 11 nodes all located in North America as shown in Figure 4. The capacity of each link is 9920Mbps, except link between Indianapolis and Atlanta, which is 2480Mbps. With this topology, we randomly choose 3 nodes to connect with clients and 2 nodes connecting with origin server.

Figure 4. Abilene Topology

The consumer behaviour is described as following: Each client will generate Interest packet with rate of 100 Interest/sec. These Interest packets refer to request contents with

total 1000 kinds of individual content, i.e., N=1000. The content popularity follows Zipf's Law with Zipf's parameter s = 0.8 [25]. For each router, the caching capacity is 10 entities, i.e., C=10. The total caching capacity in Abilene topology is thus 110 entities. Content store uses Least Recently Used (LRU) policy for content replacement. Forwarding path follows the shortest path derived by OSPF based on link information provided from Internet 2 group.

4.2 Simulation Results

We consider following indexes to evaluate the performance: Cumulative ratio of traffic served by server, hit ratio, hop reduction ratio, average link utilization, and fairness index. We use these performance indexes to compare with both non-coordinated and coordinated approaches, which are on-path caching [2] and off-path caching [4] respectively. The results are presented as follows.

First, we compare the cumulative traffic toward origin server. Normalized by total traffic, the cumulative ratio is shown in Figure 5. The horizontal axis is content ranked by popularity and the vertical axis is the cumulative traffic ratio toward server.

For on-path caching, the cumulative traffic accounts for 0.876 in total. It means that 87.6 percent of traffic is served by origin server while only 12.4 percent of traffic is served by cache nodes. In comparison with off-path caching, the cumulative traffic of our proposed method is slightly higher in initial but lower in overall. It's because we store less top-ranked popular contents in order to spare room for on-path caching. Some higher-ranking contents hence need to be acquired from origin server that results in higher traffic in the first place. However, by providing chances for cache space usage to contents with lower ranking, the cumulative traffic is lower than off-path caching at the end. The cumulative traffic ratio is 0.45 in proposed method, which is merely a half compared to on-path caching and also better than off-path caching. Cache nodes in the network lighten great amount of server loading by keeping more than a half of traffics inside network.

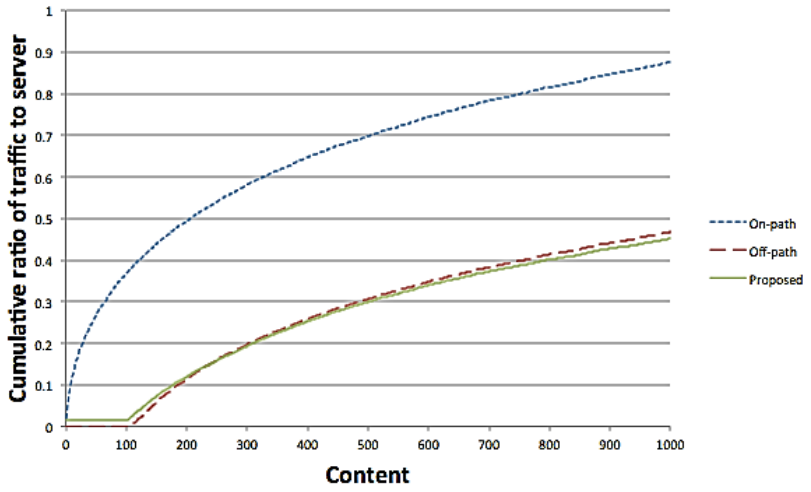

Figure 5. Cumulative Traffic Toward Origin Server Normalized by Total Traffic

Figure 6 shows the average hit ratio per content ranked by popularity. The horizontal axis is content ranked by popularity as well as vertical axis is the average hit ratio. The cache nodes running on-path caching will have higher probability to store popular contents that results in higher hit ratio of top-ranked contents. For non-popular contents ranked after 100th, there has variation of hit ratio due to smaller sample space as the number of requests decreasing with lower ranking. For off-path caching, the hit ratio of contents ranked after 110th is zero since there is no spare space remained and requests

need to be served by origin server instead. Our method maintains higher hit ratio among popular contents compared to on-path caching and among unpopular contents compared to off-path caching that leads to higher overall performance on hit ratio.

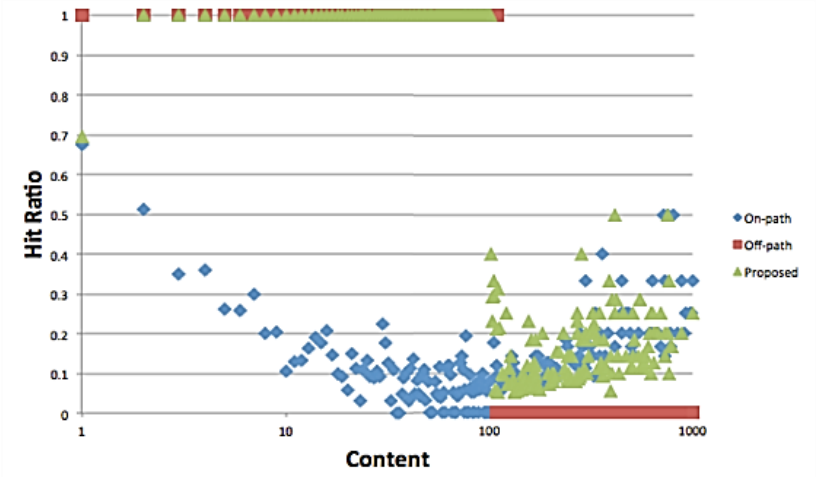

Figure 6. Average Hit Ratio per Content Ranked by Popularity

Furthermore, we derive the weighted hit ratio by multiplying hit ratio per content with its requesting frequency. The result is presented on Figure 7. For on-path caching, the weighted hit ratio is 0.124 that means a request has only 12.4 percent of likelihood to get a hit in the network. The weighted hit ratio of proposed method is 0.55 that is 4.4 times higher than on-path caching and also better than 0.53 of off-path caching hit ratio as well.

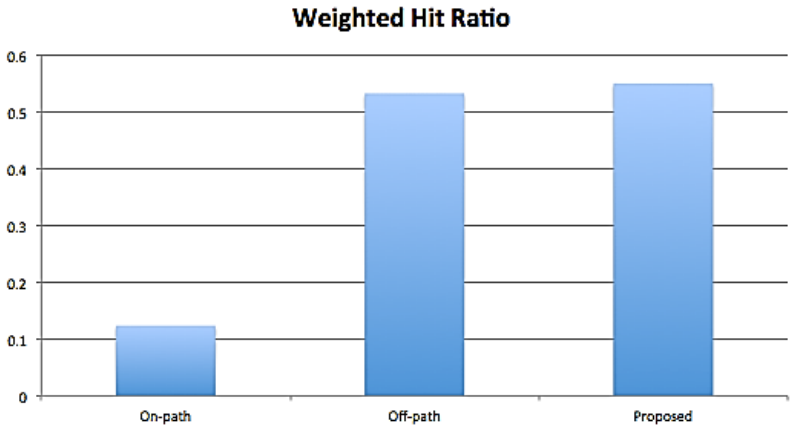

Figure 7. Weighted Hit Ratio

We present hop reduction ratio to evaluate the latency reduced by cache nodes. The hop reduction ratio is defined as (16).

$$\text{Hop reduction ratio} = \frac{H_Z - h_Z}{H_Z} \quad (16)$$

It shows the percentage of hop count decreased compared to hop count from client to server. H_Z is the average hop count number from clients to server and h_Z denotes average hop count number from clients to hit node with content Z. Figure 8 shows the result with

horizontal axis representing content ranked by popularity and vertical axis representing the hop reduction ratio per content.

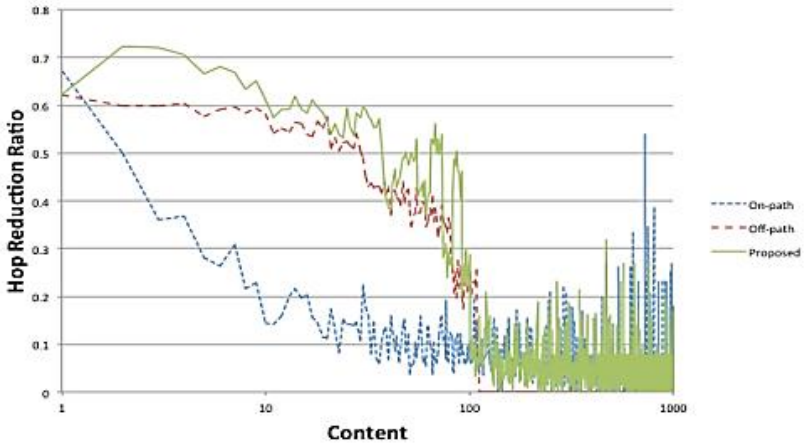

Figure 8. Hop Reduction Ratio

For on-path caching, the trend is similar with hit ratio that top-ranked popular contents have more hop reduction since they occupy most of the caching space. Compare to off-path caching, proposed method has more hop reduction on average. Among unpopular contents, it's obviously that when in off-path caching, contents ranked after 110th need to be served by origin server causing zero hop reduction. For popular content, it's more complicated. Note that, in proposed method, each router preserves part of cache space for on-path caching. It provides probability for contents to be cached right next to the node closet to every client, which is optimal for any nodes. While, in off-path caching, all clients need to fetch specific content at a single node. The cache node is chosen by average latency and thus does not guarantee optimal for every client. To sum up, the cache space preserved for on-path caching in our method helps to enhance hop reduction ratio comparing to pure on-path caching and off-path caching.

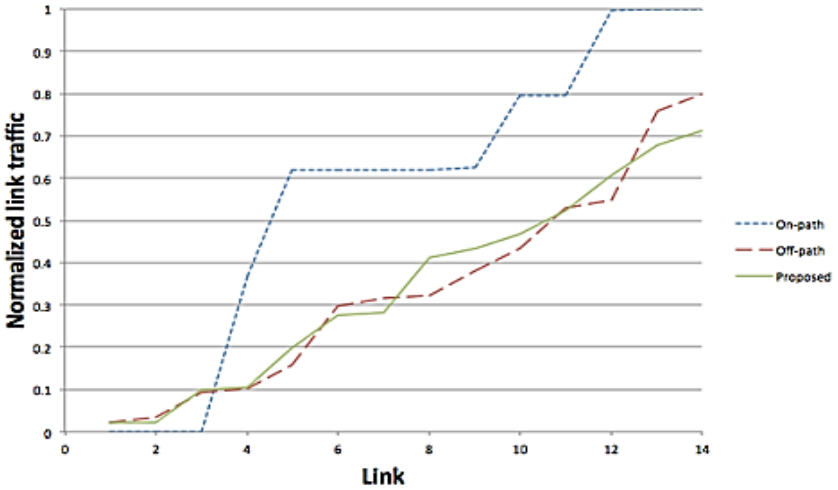

Figure 9. Normalized Link Traffic

Next, Figure 9 shows the traffic usage of each link in the network. The horizontal axis denotes link ranked by average traffic traversed on and the vertical axis denotes the link traffic normalized by maximum traffic amount. For on-path caching, a client requests contents from server by taking the single shortest path. It results that parts of links are highly utilized while other links being unused. For off-path caching, contents are stored in distributed manner so that traffic can be balanced. However, nodes containing top-ranked contents will be requested more often than other nodes and hence creates links with high utilization. Proposed method makes single node store less popular contents. It results in lightening single node's loading and the link utilization can thus be relieved. Our proposed method removes highly utilized links, i.e., bottlenecks, by offloading traffic to other links so as to balance in-network traffic.

Finally, we evaluate the fairness among all the contents by using the Jain's fairness index as (17).

$$\text{Jain's fairness index} = \frac{(\sum_{i=1}^{N} X_i)^2}{N \cdot \sum_{i=1}^{N} {X_i}^2} \tag{17}$$

The fairness here stands for the chance for each content to utilize cache space in the network. It's similar to the concept of hit ratio, which is the abstraction we use for evaluation. X_i denotes the hit ratio of i-th content ranked by popularity and N denotes the total number of contents. As Figure 10 shows, proposed method has the highest value following with on-path and off-path caching sequentially.

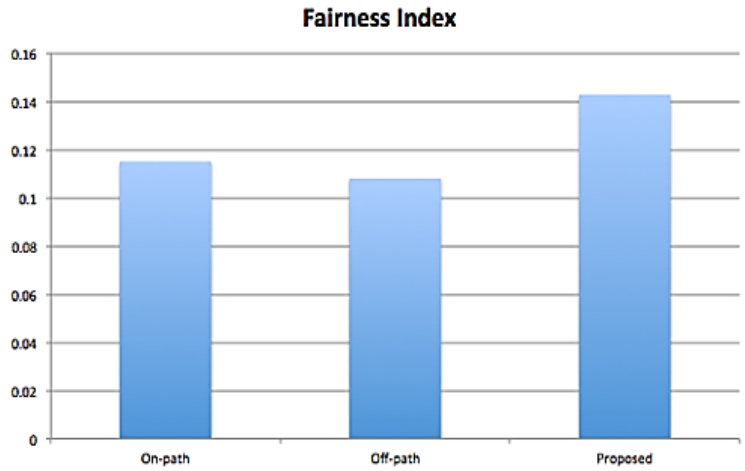

Figure 10. Fairness Index

Off-path caching is with the worst fairness index as 0.108, due to the monopoly of cache space with popular contents as well as on-path caching maintains inferior fairness with index being 0.115. Our method holds the highest fairness, with index being 0.142, by caching most popular contents while reserving space for unpopular contents as well. It's 32 percent higher than off-path caching and 24 percent higher than on-path caching.

5. Conclusion

In-network caching is an important feature of ICN. While all routers in ICN maintaining caching capability, the objects can be store in the network. It helps reduce retrieval time and save the bandwidth usage.

The default policy in most ICN approaches is on-path caching. Each object will be cached at routers on the path between requester and data source. This approach might raise the copy redundancy by storing the same object all over the network. By coordinated

caching scheme, the content placement is jointly decided based on network and router information. Most of coordinated approaches only keep the most popular contents in network. This aggressive way ignores the demand of unpopular contents and hence raises the fairness issue along with the performance declination. To overcome these problems, our study leverages the caching space for deploying on-path caching policy by taking advantages of the feature that both popular and unpopular contents can be cached in this scheme.

We propose a method to determine the fraction of caching space to be leveraged and how to place contents. The fraction is chosen for partitioning caching space and aims to minimize average latency. Then, we further consider the latency of each links between single edge router and other routers for content placement. It reduces the content redundancy while minimizing latency at the meantime. The simulation results show that, comparing to off-path caching, our approach offloads nearly two times of traffic from original content provider and brings higher hit ratio. To show the decrease of latency, with higher hop reduction rate, it takes fewer hop counts to obtain contents than both on-path and off-path caching. Besides, it removes the bottleneck in the network by offloading traffic of popular contents toward all routers. The traffic of each link is balanced so the overall link capacity can be further utilized. Finally, the fairness index shows that proposed method maintains higher fairness by giving all contents better chance to take advantage of cache space. We claim that, by leveraging both types of caching schemes together, it can achieve better performance than only deploying either one of them.

The proposed method can be applied in any ICN approaches to provide a better content delivery service. For the future direction, we are trying to deploy our method into a realistic network environment where the user behaviours are more complex in such a condition. It lets us observe and add more features from user's requesting patterns to improve our method.

References

[1] Cisco Virtual Networking Index: Forecast and Methodology, 2014-2019, White Paper **(2015)**.
[2] Jacobson, V., Smetters, D. K., Thornton, J. D., Plass, M. F., Briggs, N. H., & Braynard, R. L., "Networking named content", Proc. of ACM CoNEXT '09, **(2009)**, pp. 1–12.
[3] W. K. Chai, D. He, I. Psaras, and G. Pavlou, "Cache less for more in information-centric networks", IFIP International Federation for Information Processing, **(2012)**, pp. 27–40.
[4] Barakat, C., Kalla, A., Saucez, D., & Turletti, T., "Minimizing Bandwidth on Peering Links with Deflection in Named Data Networking", 2013 Third International Conference on Communications and Information Technology (ICCIT), **(2013)** , pp. 88–92.
[5] S. Guo, H. Xie, and G. Shi, "Collaborative forwarding and caching in content centric networks", IFIP'12 Proceedings of the 11th international IFIP TC 6 conference on Networking, **(2012)**, pp. 41–55.
[6] Li, Y., Xie, H., Wen, Y., & Zhang, Z. L., "Coordinating In-Network Caching in Content-Centric Networks: Model and Analysis", IEEE 33rd International Conference on Distributed Computing Systems, **(2013)**, pp. 62–72.
[7] Bengt Ahlgren, Christian Dannewitz, Claudio Imbrenda, Dirk Kutscher, and Börje Ohlman, "A survey of Information-Centric Networking", IEEE Communication Magazine **(2012)**, 50(7): 26-36.
[8] Koponen, T., Chawla, M., Chun, B. G., Ermolinskiy, A., Kim, K. H., Shenker, S., & Stoica, I., "A Data-Oriented (and Beyond) Network Architecture", ACM SIGCOMM Computer Communication Review, Vol. 37, No. 4, Aug. **(2007)**, pp. 181–192.
[9] Ain, M., Trossen, D., Nikander, P., Tarkoma, S., Visala, K., Rimey, K., & Kjällman, J., D2.3–Architecture Definition, Component Descriptions, and Requirements. Deliverable, PSIRP 7th FP EU-funded project, Feb. **(2009)**.
[10] Ahlgren, B., D'ambrosio, M., Dannewitz, C., Eriksson, A., Golic, J., Grönvall, B., & Mäkelä, J., "Second NetInf Architecture Description", 4WARD EU FP7 Project, Deliverable D-6.2 v2.0, Apr. **(2010)**.
[11] G. Zhang, Y. Li and T. Lin, "Caching in Information Centric Networking: A Survey", Elsevier Computer Networks, **(2013)**, pp. 3128-3141

[12] N. Laoutaris, S. Syntila, I. Stavrakakis, "Meta algorithms for hierarchical web caches", Proceedings of the 2004 IEEE International Performance, Computing and Communications, Conference, **(2004)**, pp. 445–452.
[13] Christine Fricker, Philippe Robert, James Roberts, Nada Sbihi, "Impact of traffic mix on caching performance in a content-centric network", Computer Communications Workshops (INFOCOM WKSHPS), 2012 IEEE Conference on Date of Conference, March **(2012)**.
[14] N. Laoutaris, H. Che, I. Stavrakakis, "The LCD interconnection of LRU caches and its analysis", Performance Evaluation **(2006)**, 63 (7): 609–634.
[15] S. Eum, K. Nakauchi, M. Murata, Y. Shoji, N. Nishinaga, "CATT: potential based routing with content caching for ICN", Proc. of ACM ICN '12, **(2012)**, pp. 49-54.
[16] I. Psaras, W.K. Chai, G. Pavlou, "Probabilistic in-network caching for information-centric networks", Proc. of ACM ICN '12, **(2012)**, pp. 55-60.
[17] Sung-Hwa Lim, Young-Bae Ko, Gue-Hwan Jung, Jaehoon Kim, and Myeong-Wuk Jang, "Inter-Chunk Popularity-Based Edge-First Caching in Content-Centric Networking", IEEE Communications Letters, vol. 18, No. 8, August **(2014)**, pp. 1331-1334.
[18] Yusung Kim, Ikjun Yeom, "Performance analysis of in-network caching for content centric networking", Computer Networks, Volume 57, Issue 13, September **(2013)**, pp. 2465-2482
[19] E.J. Rosensweig, J. Kurose, "Breadcrumbs: efficient, best-effort content location in cache networks", IEEE INFOCOM 2009, **(2009)**, pp. 2631-2635.
[20] Z. Li, G. Simon, "Time-shifted TV in content centric networks: the case for cooperative in-network caching", 2011 IEEE International Conference on Communications (ICC), **(2011)**, pp. 1-6.
[21] Massimo Gallo, Bruno Kauffmann, Luca Muscariello, Alain Simonian, Christian Tanguy, "Performance evaluation of the random replacement policy for networks of caches", Performance Evaluation, Volume 72, **(2014)**, Pages 16-36.
[22] Breslau, L., Cao, P., Fan, L., Phillips, G., & Shenker, S., "Web caching and zipf-like distributions: Evidence and implications", IEEE INFOCOM '99, **(1999)**, vol.1, pp. 126-134.
[23] Gill, P., Arlitt, M., Li, Z., & Mahanti, A., "Youtube traffic characterization: a view from the edge", the 7th ACM SIGCOMM conference on Internet measurement, **(2007)**, pp. 15-28.
[24] Internet2. [Online] http://www.internet2.edu/ **(2015)**
[25] Fricker, C., Robert, P., & Roberts, J., "A versaile and accurate approximation for LRU cache performance", 24th International Teletraffic Congress (ITC 24), **(2012)**, pp. 1-8.

