

A Proactive Caching Scheme Based on Content Concentration in Content-Centric Networks

Xi Luo^{1,2} and Ying An¹

¹Institute of Information Security and Big Data, Central South University, China

²Department of Information Technology, Hunan Police Academy, China

Abstract: *Content-Centric Networking (CCN) provides a new networking paradigm to overcome the challenges of the current Internet and better support the content-oriented services. In-network caching is one of the core technologies for optimizing content distribution and has been attracting ever increasing attention. In the current CCN caching schemes, the caching decision is only implemented during the content response process after the content which matches the user request is found, which lacks the capacity of proactive content advancement. This paper borrows the idea of molecular diffusion and introduces the conception of content concentration into the caching decision to analytically model the user demands for different contents in different network regions, and then a proactive caching scheme based on content concentration is proposed. In this scheme, the content replica can be proactively pushed from the high concentration region to the low concentration one to realize the fast deployment of content cache replicas and the dynamic adjustment of the cache location. Furthermore, a probabilistic content placement method is also implemented by synthetically considering the content popularity. The simulation results show that this scheme can effectively improve the cache hit ratio and shorten the delay involved in content search and response.*

Keywords: *Content-centric networking, diffusion theory, proactive caching, content popularity.*

Received October 13, 2016; accepted November 27, 2017

1. Introduction

During the recent years, with the continuous development of P2P applications, publish/subscribe systems, ubiquitous computing, and mass streaming media, information acquisition has become the main body of the current network service, and the traditional host-centric communication paradigm has gradually evolved into the information-centric paradigm. The changes in network communication require that network designers make major adjustments to the current Internet architecture, which leads to the emergence of a series of novel networks represented by the Content-Centric Networking (CCN) [9].

In CCN, the importance of the content location is gradually weakened. The contents are located and routed by their unified content names. Meanwhile, CCN requires each node to have the ability to cache the passing contents, thus providing a quick response to the follow-up requests for the same content coming from other users. In other words, CCN integrates the transparent and ubiquitous in-network caching as an inherent part of its network architecture in order to speed up the distribution of contents and improve the network resource utilization. Therefore, in-network caching has become a research hotspot in CCN and attracted considerable scholarly attention.

One of the important purposes of CCN content caching is to gradually bring the desired content closer to the user who requested it, thereby reducing the access delay to the content. However, for the entire

network, copying the desired content to the edge of the network is often a slow process [24]. The slowness can be mainly attributed to the fact that in the current CCN caching mechanisms, the caching decision is mostly driven by the users' interest, which resembles the distribution of content, and the caching location is selected from intermediate nodes on the reverse path of Interest based on an on-path caching strategy. This leads to the two main drawbacks of this method: First, the holder of the content (the original content server or intermediate caching node) only triggers decision making of cache content placement during the returning process of the response data after receiving corresponding Interest plus cache hit, which lacks the capacity to actively promote content replication. This response pattern is too passive, which slows the deployment of cache content in the network down considerably and it can be profoundly affected by the forwarding strategy of user Interest. Second, the selection of content placement location is often limited to the direct path from the original content server and the requesting user node, and the node state and user needs of the off-path nodes are not fully considered. It has been pointed out in [5] that off-path caching strategies provide more advantages than on-path strategies in performance of energy consumption and resource utilization. More importantly, if the content can be copied to the network regions outside the current delivery path beforehand based on the potential user demand in the future, the query and achievement

will be sped up further and the overall performance of the cache system greatly improved.

To solve these problems, this Paper Proposes a Proactive Caching Scheme Based on Content Concentration (PCSBCC). In this scheme, when any Interest hits in the cache of some node, the desired content will be sent back along the reverse path of the Interest, and then the placement location of the content replica can be selected on the content delivery path. This is the same as the existing on-path caching strategies. More importantly, we borrow the idea of molecular diffusion in the kinetic theory and define the Content Concentration (CC) of different net regions based on the node distribution density and the distance between the content provider and the requester. Then the content replica is actively distributed from the high concentration region to the low concentration one to realize the fast deployment of content cache replicas and the dynamic adjustment of the cache location, which improves the caching performance of the system effectively.

The rest of this paper is organized as follows. Section 2 presents some related works and section 3 introduces the idea of molecular diffusion to construct a system model. In section 4, we describe our content concentration based proactive caching scheme in detail. Section 5 demonstrates our simulation results and finally we conclude the paper in section 6.

2. Related Work

In the networking area, caching has been widely studied in the context of content distribution networks and the World-Wide Web, and the initial studies focused on the performance of different cache replacement strategies in standalone caches [1, 2, 10, 17, 20]. Due to the demand for line-speed operation of caches in CCN, however, the simplicity of algorithm is more important for cache replacement strategies. Therefore, researchers give more attention to the caching decision strategies for CCN and a lot of relevant research work has been performed in recent years.

Psaras *et al.* [21] proposed a weighted probability based caching scheme, named ProbCache. In this scheme, the requested content is replicated at each node with some probability. The probability is inversely proportional to the distance between the current node and the requester, and proportional to the available resources of the node. It ensures that the contents have more chances to be cached at the nodes closer to the requester. In this way, it can quickly push the content copies to the network edge and meanwhile improve the caching performance. A similar strategy was used in three other studies, in which a longer cache time was established for more popular content to ensure that their cache hits would be successful and foster a faster response time [13, 16, 19]. A

differentiated caching scheme was studied in yet another study [7]. This method involved a two-dimensional differential caching strategy based on the caching location and cache-resident time. In this strategy, the number of requests for certain specific content was defined as the content activity. According to the change of content activity, the caching location is pushed downstream hop by hop in the spatial dimension in order to spread popular contents to the network edge in a gradual manner. Liu *et al.* [14] proposed an a content-Aware Placement, Discovery and Replacement (APDR) mechanism which integrates the placement, discovery and replacement to realize an orderly cache for the contents. The main idea behind APDR can be described as follows: the Interest packet carries the request for content and collects information about the potential user demand and the available cache of the nodes along the path, based on which the destination of Interest makes caching decision and attaches the decision information to the Data packet to indicate the time that the requested content will be cached at different nodes. In [15], a Selective Caching (SC) mechanism was presented according to the user's potential demands for content and the content popularity. According to the author, due to the existence of the downstream node cache, the port that responded to a received request for certain content will not receive requests for the same content again. For this reason, for certain content, when fewer ports fail to receive requests for specific content, the potential demand for that content is considered low and therefore is believed to be less need for the node to cache it. The mechanism underlying this process, as given above, involves combining factors such as the link utilization rate and the cache space of the nodes to calculate the probability of a content being cached and implement selective caching of the desired content. In this way the cache efficiency of the system is improved. Thar *et al.* [23] proposed a cooperative caching decision mechanism based on consistent hashing. In this method, the routing nodes within AS were divided into different groups. Then the caching performance and resource utilization can be enhanced by cooperative content caching and request forwarding among the nodes in the group. For caching decisions, the content popularity was the primary factor considered and consistent hashing was used to prevent the redundant caching of identical contents. In addition, other studies such as [4, 11, 22] seek to design the caching strategies for CCN by using the social attributes of network users.

Some existing research works have tried to optimize the problem of caching decision in CCN by introducing the concept of physical field. In one such study, a potential-based forwarding scheme with random caching is proposed [6]. In this method, the concept of "potential value" was introduced for each node to guide the Interest packets from the node with

low potential value toward the node with high potential value. And then the Data packet will be randomly cached at one node on the reverse path. However, it does not present any concrete method to choose proper potential value, so the actual performance cannot be guaranteed. Ge *et al.* [8] proposed a hybrid routing scheme based on differential cache advertisement, which is inspired by the idea of data field. It pointed out that nodes capable of responding to the requests for contents can be divided into two categories, stable content servers and volatile caching nodes. Considering the differences between these two categories of nodes, the globally oriented potential field and locally attracting potential field are constructed respectively to realize global routing and local response to the content request. In this way, it tries to improve the overall utilization of cache resources. In [3], a topological potential based caching scheme was presented. It constructed a content-based topological potential field to guide the distribution of cached content advertisements and provided support for the node mobility. However, these methods were all designed to improve the caching performance and network communication service quality from the perspective of optimizing the forwarding process of cached content advertisements or user requests.

Unlike the methods mentioned above, the method proposed here constructs the content concentration in the network and realizes proactive content placement and dynamic content migration by taking advantage of the diffusion of content replicas between regions with different content concentrations, thus increasing the possibility of providing nearest response to a potential user request. The proactive advancement pattern of content replica will greatly shorten the delay involved in content search and response and improve the probability of a cache hit.

3. Introduction of Molecular Diffusion

Diffusion is the net movement of molecules or atoms from a region of high concentration to a region of low concentration, until a uniform distribution is reached. According to the kinetic theory, molecular diffusion is in essence a mass transfer of molecules caused by thermal motion. From the microscopic view, it is actually the result of collision between molecules when large numbers of molecules are in random thermal motion. Since the molecules are distributed unevenly in different spatial regions, the collisions force molecules to move from areas of high distribution density to those of low distribution density before finally reaching uniform distribution. Similar phenomena can be observed frequently in nature. For example, species often gradually disperse to search for suitable habitat due to local overpopulation and scarcity of living resources. Many biological and non-biological systems also make use of diffusion.

Examples include air exchange during breathing, when oxygen travelling from the alveolus into the blood and carbon dioxide from the blood into the alveolus; diffusion of impurity atoms in silicon chips. The spread of germs and proliferation of information also resemble diffusion.

The diffusion law, also known as the Fick's law, describes the mass transfer process caused by molecular diffusion, which was first reported in 1855 by German physicist A. Fick based on experimental results. According to Fick's law, during the process of steady state diffusion, the diffuse flux J is proportional to the negative of the concentration gradient, which can be described as follows.

$$J = -D \frac{\partial \phi}{\partial x} \quad (1)$$

Here, D is the diffusion coefficient, an important physical parameter describing the diffusion velocity. It describes the substance flow passed through the unit cross section per unit of time under the condition of one unit concentration gradient. ϕ is the concentration and x means the position. The minus sign in the formula indicates that substance can diffuse along the direction of decreasing concentration. As shown in Equation (1), flow (re-distribution) is caused by non-uniform spatial distribution of substance, and the resulting substance flow is proportional to the gradient of the substance's concentration, during which gradient is the driving force of substance flow.

Inspired by the idea of diffusion of molecules, the content object is here treated as solute and the user demand as the solvent, so the whole network system can be considered a solution containing the contents and users' demands. Due to the uneven and dynamic distribution of contents and user demands, the concentration of the solution also varies among regions, with the concentration of each type of content higher near its original server or caching node and lower around the request node. Intuitively, as far as caching decisions, content replica should try to approach the region with higher user demands so as to provide fast and efficient caching and communication services. For this reason, the caching decision is here considered the diffusion process of solute molecules (content replicas) from regions of high concentration to regions of low concentration.

4. PCSBCC Caching Scheme

4.1. Model of Content Diffusion

Assume that in the network there is a set of content $C = \{c_i | i = 1, 2, \dots, k\}$, in which the elements c_i are numbered according to the popularity level and the corresponding access probability is represented with p_i . For any $1 \leq i \leq j \leq k$, there goes $p_i \leq p_j$. p_i is considered the popularity of the content c_i in the set of content C , and the set $P = \{p_i | i = 1, 2, \dots, k\}$ is regarded

as the popularity distribution of C . Previous work has shown the popularity of contents follows Zipf or Zipf-like distribution [12]. Therefore, the popularity of any element c_i in C can be described as follows:

$$P_i = \frac{\Omega}{i^\alpha} \quad (2)$$

In which, $\Omega = \left(\sum_{i=1}^k \frac{1}{i^\alpha} \right)^{-1}$.

If the contents with different levels of popularity are considered different types of solutes, then each copy of certain content can be considered one molecule of the solute. The cache placement process of certain type of content c_i can be seen as the diffusion process of solute molecules c_i in solvents. In this way, we construct a content diffusion model to describe the motion of content replicas during the caching process in CCN. The concepts of diffusion coefficient and content concentration are redefined as follows.

- **Definition 1: Diffusion Coefficient (D).** The diffusion coefficient is one of the physical properties which are used to indicate the diffusion capacity of a substance. In the process of caching, the popularity of content reflects the demand from the users. More popular content requires more cached replicas to meet user demands, which means that such contents are of higher diffusion potential to ensure that its cached replicas can be distributed promptly and densely to the network. Therefore, we used content popularity as the diffusion coefficient of content c_i :

$$D_i = P_i = \frac{\Omega}{i^\alpha} \quad (3)$$

- **Definition 2: Content Concentration (CC).** For the content c_i in a given node N , its content concentration is defined as the degree that the cached replicas of c_i satisfy the users' demands around the node N . In our diffusion model, the content concentrations of content c_i in its original content server and its caching nodes are set to the maximum value of 1. And the content concentrations in the requesting nodes of c_i are set to the minimum value of 0. Then for any location of intermediate node N , the content concentration of c_i should be proportion to the distance from N to the nearest location of cached replica for c_i (d_{NC}) and inversely proportion to the distance from N to the nearest requesting node of c_i (d_{NR}). For certain content, the existence of more user nodes within a certain region may imply a high probability of sending requests for such content in the future by users within that region. For this reason, we use the degree centrality of node N , D_{CN} , to express the density of nodes within corresponding region, which is in inverse proportion to the content concentration. Then the $CC_N^{c_i}$ can be expressed as follows:

$$CC_N^{c_i} = \begin{cases} e^{-\frac{d_{NC} \cdot D_{CN}}{d_{NR}}}, & d_{NR} > 0 \\ 0, & d_{NR} = 0 \end{cases} \quad (4)$$

4.2. Working Principle of PCSBCC

The basic idea underlying PCSBCC is to overcome the disadvantages of the existing caching mechanisms that the decision-making process is solely driven by the Interest packets of users. By using the idea borrowed from molecular diffusion, PCSBCC provides a more proactive way that can rapidly push the content replica to the user side. At the same time, the content concentration and the content popularity are considered during the advancement process of content replicas to realize a probability-based cache placement, which accelerates the response to subsequent user requests and improve the overall caching performance of the system. PCSBCC has two main parts: advancement of cached content and caching decision for the content. During the process of cached content advancement, the replica of cached content can be advanced by two basic patterns: passive response and proactive diffusion. The passive response pattern occurs when some node returns the requested data, which means that the node would send back a copy of the requested content along the reverse path of the Interest packet only after a corresponding user request is received and a hit occurs in its cache. By contrast, the trigger of the proactive diffusion pattern is not relevant to whether a cache hit is occurred. The node calculates and compares its content concentration to that of its neighbouring nodes at regular intervals (what we call diffusion period, T_d), and then diffuse the copy of content hop by hop along the path in which the content concentration of cached content decreases most rapidly. No matter which advancement pattern is adopted, the cache location is selected from the advancement path of content replicas with a certain probability. The main procedure of PCSBCC is described as follows.

- **Step 1:** A timer (T) is contained in each node and the timer is reset and starts to count time upon network initialization. In case any node N contains a cached content set C that is not empty (at least cached one content object in the node), then certain content c_i is picked out as the pending object based on the popularity with a certain probability by N .
- **Step 2:** $CC_N^{c_i}$ (the content concentration of the cached content c_i at current location) is calculated by N according to Equation (4) and is compared with the values calculated by N 's neighbors. If $CC_N^{c_i}$ is larger than the content concentration in N 's neighbouring region, then the replica of c_i will be actively passed to the neighbor node in the steepest descent direction of content concentration and the timer of node N is reset. Otherwise, the proactive diffusion does not happen at node N .

- *Step 3*: Step 1 and Step 2 are repeated every diffusion period to determine if proactive diffusion should be performed.
- *Step 4*: When node N receives a replica of content c_j , N will cache the content replica with certain probability based on the probabilistic caching strategy. The Pending forwarding Table (PIT) is checked at the same time.
- *Step 5*: If the user request matching c_j is found in the PIT of N, then the passive response pattern is triggered by N, and c_j will be sent out the face the Interest arrived on to respond to user requests. Otherwise, N will choose c_j as the pending object and execute Step 2 to decide whether the proactive diffusion pattern should be triggered.
- *Step 6*: If c_j is outside the cache of node N and neither the proactive diffusion nor passive response is triggered, then c_j will be directly rejected by N.

Here, the main purpose of proactive diffusion is to achieve the preliminary advancement and deployment of the content replica and reduce the delay of content search and retrieval. The implementation details of PCSBCC will be explained in the following sections.

4.2.1. Calculation of Content Concentration

In the PCSBCC scheme, the proactive diffusion of content replica and caching probability are closely related with content concentration. As shown in Equation (4), the content concentration $CC_N^{c_i}$ for content c_i at node N is determined by d_{NC} , the distance between N and nearest copy of content c_i , and d_{NR} , the distance between N and the nearest request node for c_i . In the real network, however, the location of request node is usually unable to be predicted and the location of cached content is also uncertain. That is to say, d_{NC} and d_{NR} may both be variable. Therefore, the value of $CC_N^{c_i}$ should be updated according to the change of d_{NC} and d_{NR} . Obviously, d_{NC} equals zero at the holder of c_i (the original server or the intermediate caching node), and d_{NR} equals zero at the requester of c_i . Under other circumstances, we assumed that the initial values of d_{NC} and d_{NR} to be relatively large values (for example the total number of nodes in the network). In order to acquire the distance information to the content requester and the content holder, a hop count field (initial value is equal to 0) is added in both the Interest packet and Data packet to record the number of hops with forwarding from the original source node to the current location. When certain Interest packet is received by a node, the hop count contained in the packet header will be compared to the current d_{NR} so that d_{NR} will be updated with the smaller value of the two. Similarly, the node updates the value of d_{NC} according to the hop count contained in the Data packet. Each node calculates the content concentration

of specific content according to Equation (4) and updates d_{NC} , d_{NR} , or degree centrality when necessary.

4.2.2. Proactive Diffusion of Content

The trigger of passive response pattern in PCSBCC is relatively simple. It is triggered by the Interest packet which matches the cached content and the satisfied content is forwarded along the reverse path of the Interest packet. In the proactive diffusion pattern, the diffusion of content replicas is driven by the content concentration difference of cached content between the adjacent nodes and the content replica moves from the node with higher concentration to the node with lower concentration. Unlike the passive response pattern, where both the forwarded content and its forwarding path are determined by the corresponding Interest packet, the proactive diffusion pattern mainly addresses the following two problems:

1. There may exist many different contents in the node cache, the diffusion of all cached contents easily causes extensive communication and storage cost, so a reasonable selection of the diffused content must be made;
 2. Diffusing contents in all directions will essentially produce a large number of redundant content replicas that excessively consume network resources. In this way, a selective content diffusion must be also realized to reasonably select the advancement direction of the content replicas. More specific solutions to these two problems are given as follows.
- Selection of contents to be diffused In order to ensure the fairness among different contents and the diversity of the cached content, the content to be proactively diffused is selected from the cache with probability proportional to its popularity. Specifically, assume that there are l contents in the cache of node N, the i -th content object and its popularity can be expressed with c_i^N and $p_{c_i^N}$, respectively. Let DP_i denote the probability that c_i^N is selected as the diffused content, then it can be calculated using Equation (5).

$$DP_i = \frac{p_{c_i^N}}{\sum_{i=1}^l p_{c_i^N}} \quad (5)$$

- Selection of diffusion direction In order to control the transmission overhead of content diffusion, the node will compare its calculated value of the corresponding content concentration to those of its neighbors, and then advance the selected content to the neighbor node in the steepest descent direction of content concentration. Take node U as an example: assume that the set of neighbors of U can be described with $Y=\{N_1, N_2, \dots, N_n\}$, and the selected content object to be diffused is denoted by c_i . Then the content concentration difference of c_i

between node U and any neighbor node N_j can be described as follows:

$$\Delta CC_{uj}^{c_i} = CC_u^{c_i} - CC_j^{c_i} \quad (6)$$

If there exist a node N_i , which is the neighbor of U, such that $\Delta CC_{ui}^{c_i} = \text{Max}\{\Delta CC_{uj}^{c_i} \mid j = 1, 2, 3, \dots, n\}$ and $\Delta CC_{ui}^{c_i} > \theta_{th}$, then node U will decide to diffuse c_i toward node N_i . Otherwise, the proactive diffusion of c_i will not be triggered by node U. Here, θ_{th} is termed as diffusion concentration difference threshold, which is a positive number and expresses the minimum content concentration difference between nodes required for proactive diffusion. It is used to prevent the unnecessary diffusion resulted from the oscillation of content concentration between nodes, which may cause the algorithm cannot achieve convergence.

4.2.3. Probabilistic Content Caching

In the PCSBCC scheme, replicas of content will be cached with a certain probability by the nodes along the forwarding path of the content replica. Two main factors are considered to determine the probabilities of content replicas being cached by different nodes. First of all, from the perspective of the content itself, higher content popularity leads to higher probability of being requested by the users, which means that such content should be cached with a higher probability by nodes. However, from the node's point of view, to cache the content at the node near to the requester can speed up the response and increase the cache hit rate. The caching probability of content should increase hop by hop along the forwarding path of content, and its increment should be proportional to the decrement of the content concentration.

Specifically, assume node S is an original content server or a caching node for content c_i , and the popularity of c_i is expressed with p_i . The advancement path of c_i is described as $path_i = \{N_1, N_2, N_3, \dots, N_k, \dots\}$, and N_k is the k -th node on $path_i$. Let $CC_k^{c_i}$ denote the content concentration for content c_i at N_k and $\Delta CC_{s1}^{c_i}$ present the content concentration difference between S and the first node N_1 , then the probability for N_k to cache c_i can be expressed as follows:

$$CP_k^{c_i} = \min \left\{ p_i \cdot \frac{1 - CC_k^{c_i}}{\Delta CC_{s1}^{c_i}}, 1 \right\} \quad (7)$$

It should be noted that once N_k becomes the new caching node for content c_i , then N_k will be regarded as the new starting node and the subpath following N_k will be the new forwarding path for content c_i . Then the caching probability for c_i at each node along the path will be calculated with Equation (7) and the probabilistic content caching can be implemented.

5. Performance Simulation and Evaluation

We conducted simulations using the ndnSIM [18], a NS-3 based simulator which is specially designed for Named Data Networking (NDN) implementation. In our simulations, a flat random network topology is constructed by the locality model of the Georgia Tech Internetwork Topology Models (GT-ITM) topology generator, which consists of 2 content servers and 50 routing nodes. The servers located at the edge of network and 2 edge nodes are selected randomly to connect to the servers respectively. The contents stored in each server are selected randomly from the content pool in a non-repeated way and its quantity accounts for half the total content objects in the network. It is assumed that the user request pattern follows a Zipf distribution with parameter α and the content request arrivals follow a Poisson process with arrival rate $\lambda=50$ requests per second. Each routing node has a cache of the same size and the initial cache state of each node is empty. The Interest packet is transmitted by flooding and the default cache replacement policy is Least Recently Used (LRU). The remaining major experimental parameters are set as given in Table 1 except special declaration.

Table 1. Experimental parameter settings.

Major parameters	Default	Range
Number of content	5000	1000~10,000
Size of content(MB)	1	
Cache size of node (MB)	50	10~100
User request pattern	Zipf: $\alpha=0.7$	0.1~1.0
Simulation time (sec)	200	

In order to evaluate the performance of PCSBCC, the cache hit rate and average access cost are selected as the main performance metrics.

- Cache hit rate the cache hit ratio is one of the most important metrics used to evaluate the caching performance. It is defined as the probability that a request is responded by the caching nodes instead of the original content servers. The higher the cache hit rate, the smaller the response rate and the load of the original content server and the higher the efficiency of the cache system.
- Average access cost Average access cost is defined as the average hop count taken by user Interest packets to meet the satisfied content objects. It reflects the speed of response to the user's request. The lower average access cost indicates higher access speed as well as higher system efficiency.

5.1. Selection of Parameters for PCSBCC

Two main algorithmic parameters are contained in PCSBCC: θ_{th} , the diffusion concentration difference threshold, and T_d , the diffusion period. In this section, we choose the appropriate values for the two parameters through simulation experiments.

5.1.1. Diffusion Concentration Difference Threshold

The relative relationship between content concentration and θ_{th} is the trigger for proactive diffusion of content in PCSBCC. A smaller θ_{th} means a higher probability for proactive diffusion of content being triggered by the node and a wider diffusion range of content in the network, which promotes the advancement and placement of content replicas in the network. However, the increase of proactive diffusion also brings greater costs. Excessively small values of θ_{th} may result in excessive diffusion and caching of content replicas, and overconsumption of the node cache inevitably leads to excessive cache replacement, which decreases the cache hit rate and limits the advantage of PCSBCC in improving response speed. Conversely, larger values of θ_{th} will reduce the proactive diffusion of content. Along with the increase of θ_{th} , the positive effects of PCSBCC on caching performance will be weakened gradually. Figure 1 shows the effects of θ_{th} on cache hit rate and average access cost. It can tell from the figure that when the value of θ_{th} is around 0.02, PCSBCC has relatively high cache hit rate and low average access cost. Up-regulation or down-regulation of θ_{th} both reduces the performance of PCSBCC in varying degrees. When θ_{th} grows over 0.02, the advantage that brought about by proactive diffusion of content decreases continuously and finally PCSBCC degenerates into the pure passive response pattern. When θ_{th} decreases continuously from 0.02, a sharp drop in the cache hit rate and a rapid increase of average access cost were detected. This is because that the overly aggressive content diffusion increases the resource consumption of node, it inevitably triggers frequent cache replacement that greatly shortens the caching service time at the node and finally weakens the caching performance of PCSBCC. Based on the results of simulation experiments and the above analysis, θ_{th} is set to 0.02 in this paper.

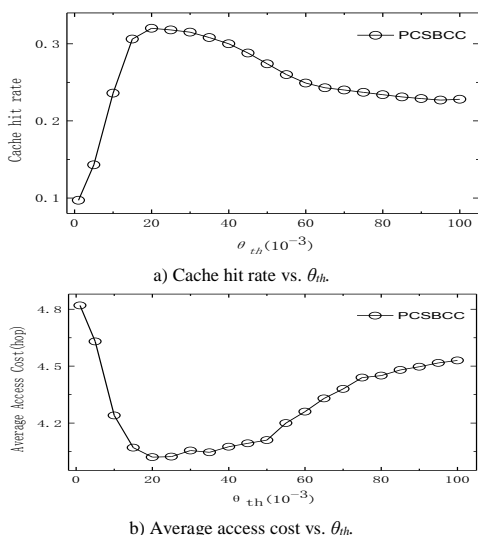


Figure 1. Impact of θ_{th} on caching performance.

5.1.2. Diffusion Period

In PCSBCC scheme, the carrier of content performs proactive diffusion according to the content concentration difference with its neighbors, and the diffusion period T_d is one of the important factors affecting the performance of PCSBCC. To reasonably choose the length of the diffusion period, we observe the performance of PCSBCC with different values of T_d through simulations. In the simulations, θ_{th} is set to 0.02 and the other experimental parameters use the default values. As shown in Figure 2, when T_d is relatively small, the calculations and comparisons for content concentration are performed by the nodes much more frequently. It may cause excessive diffusion of content replicas and greatly increase the network overhead of PCSBCC. It can be seen from Figure 2-a that the cache hit rate is only 29.1% with $T_d = 10$ ms. When T_d increases to 100 ms, the cache hit rate of PCSBCC gradually is improved to 32%. As T_d increases from 100 ms to 300 ms, however, a significant decline of cache hit rate can be seen. Such changes are mainly caused by the decrease of proactive diffusion from the node with further increase of T_d , which slows down the advancement and cache placement of content in the network and weakens the benefit of proactive diffusion considerably. The changing trend from decreasing to increasing of average access cost in PCSBCC shown in Figure 2-b can be explained in a similar fashion, whose turning point is observed around $T_d = 100$ ms, indicating where the optimal outcome is achieved. Therefore, considering the results of simulation experiment, the diffusion period T_d used in this paper is set to 100 ms.

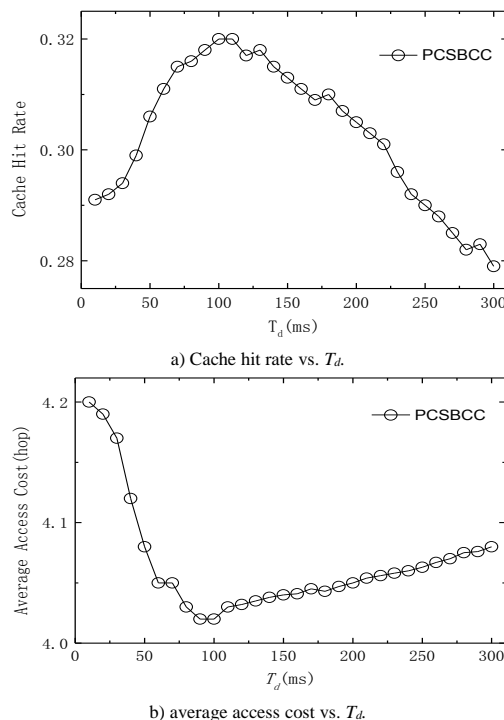


Figure 2. Impact of T_d on caching performance.

5.2. Performance Evaluation

During the simulation, the PCSBCC is compared to Betw (which uses the pure passive response pattern and only chooses the nodes having the highest betweenness centrality along the delivery path as the caching nodes) and CATT (which uses the potential field to actively carry out local content advertisement). The advertisement range in CATT is set to 2 hops. The performances of the three caching schemes are evaluated with different cache sizes, different content numbers and different values of Zipf parameter α , respectively.

5.2.1. Impact of Cache Size

The impact of cache size on the caching performance of the above listed caching schemes is investigated in this subsection. Figure 3-a shows that with the increase of the node cache space, nodes can keep the cached contents much longer, thus the chance for cache hits increases accordingly. Because of the proactive diffusion of content and the probabilistic caching decision based on content concentration and popularity, PCSBCC can spread contents near to the users more quickly and efficiently, so it achieves the highest cache hit rate among the three schemes. The advancement of content replicas in Betw is achieved by the passive response pattern completely and the selection of caching node merely depends on the betweenness centrality of nodes. It causes the high betweenness nodes suffer excessive cache load and frequent cache replacements. CATT uses potential field to realize the active advertisement of cached contents and effectively guide the user requests toward the matched content replicas, which improves the cache hit rate of the system to a certain extent. It explains why the cache hit rate of CATT is slightly higher than that of Betw.

With the increase of cache size, the caching capacity of intermediate nodes is enhanced and each content packet can be cached for a longer time. This means that users have more chance to obtain content rapidly from the nearest intermediate caching nodes. In this way, it can be seen from Figure 3-b that the average access cost of each mechanism decreases with the increase of node cache. Owing to the proactive diffusion of content in PCSBCC, the contents can be pushed to the corresponding requesters quickly. Therefore, PCSBCC obtains the least average access cost of three. When the cache size is 50MB, the average access cost of PCSBCC is only 4.02 hops, which is about 11.6% and 4.7% lower than that of CATT and Betw, respectively.

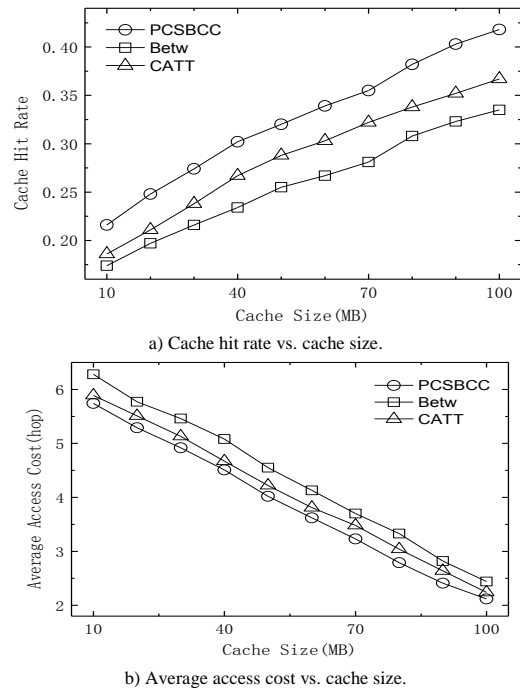


Figure 3. Impact of cache size on caching performance.

5.2.2. Impact of Content Number

The impact of content number on caching performance is discussed here, and the results are shown in Figure 4. Since when the cache size is fixed, cache resources will become much scarcer as content number increases. For that reason, it is yet another demonstration of the relationship between node resources and caching performance.

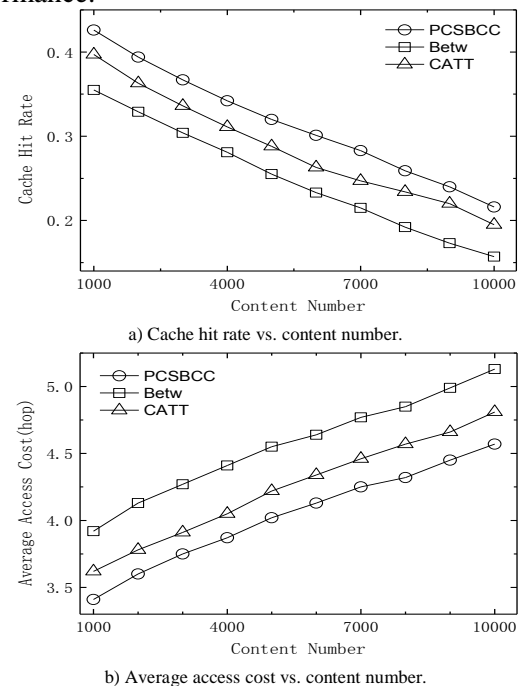


Figure 4. Impact of content number on caching performance.

Figure 4-a shows that the cache hit rate of each scheme declines significantly when the content number increases. However the cache hit rate of PCSBCC is the highest of the three, which is up to 21.6% with the content number increasing to 10,000, while the cache

hit rates of CATT and Betw only reach 19.5% and 15.7%, respectively. As shown in Figure 4-b the average access costs of the three schemes increase gradually with the increase of content number, and the average cost of PCSBCC is still significantly lower than the other two schemes. When the content number is 10,000, for example, the average access cost of PCSBCC is only 4.57 hops, which is approximately 10.9% and 5% lower than that of Betw and CATT, respectively.

5.2.3. Impact of Zipf Parameter α

As is known, users may have different preferences for the requesting contents. In this subsection the impact of user preferences on the performance of different schemes is investigated further. A larger Zipf parameter α means that there is more user requests issued for the popular contents. Additionally, the caching strategies adapted in all three schemes are inclined to give priority to more popular contents. Thus, the cache hit rate increases with α as indicated in Figure 5-a. PCSBCC achieves the best performance on cache hit rate, which benefits from the consideration of content popularity in content advancement and caching decision. The results shown in Figure 5-b also indicate that, there is a significant decline in the average access cost of each scheme as the increase of α . Because of the active advancement pattern of content, PCSBCC can effectively increase the content acquisition speed. Therefore, it obtains the least average access cost of three.

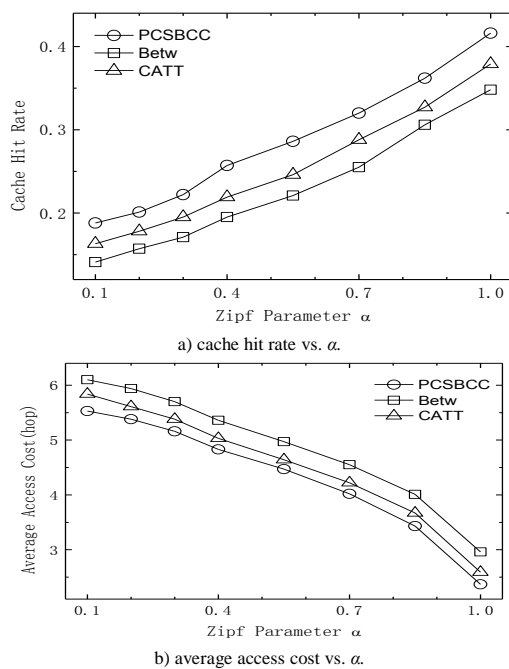


Figure 5. Impact of Zipf parameter α on caching performance.

6. Conclusions

In order to speed up the advancement of content and further improve the chances for providing the nearest

response to the potential user requests in CCN, the concept of content concentration is introduced and a content concentration based proactive caching scheme is proposed, which is inspired from the idea of molecular diffusion. The relative relationship between content concentrations at different nodes is used to trigger the proactive advancement and migration of content replicas in the network, and the probabilistic caching decision is realized by considering factors such as content popularity. Simulation results show that the proposed scheme can effectively improve the overall caching performance of the system. In the future, we plan to explore the efficiency of PCSBCC in the context of more realistic applications and extend our algorithm to the mobile network and other complex network environment.

Acknowledgement

This work is supported by the National Natural Science Foundation of China under Grant No. 61402541, and the Scientific Research Fund of Hunan Provincial Education Department under Grant No. 15B127.

References

- [1] Abdelfattah A. and Samra A., "Least Recently Plus Five Least Frequently Replacement Policy (LR+5LF)," *The International Arab Journal of Information Technology*, vol. 9, no. 1, pp. 16-21, 2010.
- [2] Balamash A. and Krunz M., "An Overview of Web Caching Replacement Algorithms," *Communications Surveys and Tutorials*, vol. 6, no. 2, pp. 44-56, 2004.
- [3] Chen L., Tang H., and Zheng L., "Consumer Mobility Scheme in Content Centric Networks Based On Topological Potential," *Computer Engineering and Design*, vol. 35, no. 8, pp. 2685-2690, 2014.
- [4] Cheng Y., Wu M., Zhao M., and Wang K., "Socially-aware NodeRank-based Caching Strategy for Content-Centric Networking," in *Proceedings of the 13th International Symposium on Wireless Communication Systems*, Poznań, pp. 297-303, 2016.
- [5] Draxler M. and Karl H., "Efficiency of on-Path and Off-Path Caching Strategies In Information Centric Networks," in *Proceedings of IEEE International Conference on Green Computing and Communications (GreenCom)*, Besancon, pp. 581-587, 2012.
- [6] Eum S., Nakauchi K., Murata M., Shoji Y., and Nishinaga N., "CATT: Potential Based Routing with Content Caching for ICN," *IEEE Communications Magazine*, vol. 50, no. 12, pp. 60-67, 2012.
- [7] Ge G., Guo Y., Liu C., and Lan V.,

- “Collaborative Caching Algorithm Based on Request Correlation in Named Data Networking,” *Journal of Electronics and Information Technology*, vol. 36, no. 12, pp. 2795-2801, 2014.
- [8] Ge G., Guo Y., Liu C., and Lan V., “A Hybrid Routing Scheme Based on Differentiated Cache Advertisement in Content Centric Networking,” *Journal of Electronics and Information Technology*, vol. 37, no. 3, pp. 700-707, 2015.
- [9] Jacobson V., Smetters D., Thornton J., Plass M., Briggs N., and Braynard R., “Networking Named Content,” *Communications of the ACM*, vol. 55, no. 1, pp. 117-124, 2012.
- [10] Jelenković P., Radovanović A., and Squillante M., “Critical Sizing of LRU Caches with Dependent Requests,” *Journal of Applied Probability*, vol. 43, no. 4, pp. 1013-1027, 2006.
- [11] Khan J. and Ghamri-Doudane Y., “SAVING: Socially Aware Vehicular Information-Centric Networking,” *IEEE Communications Magazine*, vol. 54, no. 8, pp. 100-107, 2016.
- [12] Kim Y. and Yeom I., “Performance Analysis of In-Network Caching for Content-Centric Networking,” *Computer Networks*, vol. 57, no. 13, pp. 2465-2482, 2013.
- [13] Li Y., Zhang T., Xu X., Zeng Z., and Liu Y., “Content Popularity and Node Level Matched Based Probability Caching for Content Centric Networks,” in *Proceedings of IEEE/CIC International Conference on Communications*, Chengdu, pp. 1-6, 2016.
- [14] Liu W., Yu S., Cai J., and Gao Y., “Scheme for Cooperative Caching in ICN,” *Journal of Software*, vol. 24, no. 8, pp. 1947-1962, 2013.
- [15] Liu W., Yu S., Hu X., and Zhu P., “Selective Caching in Content-Centric Networking,” *Chinese Journal of Computers*, vol. 37, no. 2, pp. 275-288, 2014.
- [16] Liu Y., Zhu D., and Ma W., “A Novel Cooperative Caching Scheme for Content Centric Mobile Ad Hoc Networks,” in *Proceedings of the IEEE Symposium on Computers and Communication*, Messina, pp. 824-829, 2016.
- [17] Ma Y., Liu X., Zhang S., Xiang R., Liu Y., and Xie T., “Measurement and Analysis of Mobile Web Cache Performance,” in *Proceedings of the 24th International Conference on World Wide Web*, Florence, pp. 691-701, 2015.
- [18] Mastorakis S., Afanasyev A., Moiseenko I., and Zhang L., “ndnSIM 2.0: A New Version of the NDN Simulator for NS-3,” Technical Report, NDN-0028, 2015.
- [19] Ming Z., Xu M., and Wang D., “Age-Based Cooperative Caching in Information-Centric Network,” in *Proceedings of the 23rd International Conference on Computer Communication and Networks*, Shanghai, pp. 1-8, 2014.
- [20] Podlipnig S. and Böszörmenyi L., “A survey of Web Cache Replacement Strategies,” *ACM Computing Surveys*, vol. 35, no. 4, pp. 374-398, 2003.
- [21] Psaras I., Chai W., and Pavlou G., “Probabilistic In-Network Caching for Information-Centric Networks,” in *Proceedings of the 2nd edition of the ICN Workshop on Information-Centric Networking*, Helsinki, pp. 55-60, 2012.
- [22] Sirichotedumrong W., Kumwilaisak W., Tarnoi S., and Thatphithakkul N., “Prioritized Probabilistic Caching Algorithm in Content Centric Networks,” in *Proceedings of the 12th International Conference on Computing and Information Technology*, Khon Kaen, pp. 255-265, 2016.
- [23] Thar K., Ullah S., and Hong C., “Consistent Hashing Based Cooperative Caching and Forwarding in Content Centric Network,” in *Proceedings of the 16th Asia-Pacific Network Operations and Management Symposium*, Hsinchu, pp. 1-4, 2014.
- [24] Zhang G., Li Y., Lin T., and Tang H., “Survey of In-Network Caching Techniques in Information-Centric Networks,” *Journal of Software*, vol. 25, no. 1, pp. 154-175, 2014.



Xi Luo is a lecturer in Department of Information Technology, Hunan Police Academy, P. R. China. She received her B.S. and M.S. degree in Communications Engineering from Central South University, P. R. China, and her Ph.D degree in computer science from Central South University, P. R. China. Her general research interests include content-centric networks, in-network caching technology and content distribution.



Ying An is an associate professor in Institute of Information Security and Big Data, Central South University, P. R. China. He received his B.S. degree in automatic control from Central South University, P. R. China, and his M.S. degree in communications engineering from Central South University, P. R. China, and his Ph.D degree in computer science from Central South University, P. R. China. His research interests include information centric networks, delay tolerant networks and optimization of future internet.