UNIVERSITY OF WEST LONDON
The Career University

UWL REPOSITORY

repository.uwl.ac.uk

Decision-making for in-network caching of peer-to-peer content chunks: an analytical modelling study

Zhang, Xu, Wang, Ning, Vassilakis, Vassilios and Howarth, Michael (2015) Decision-making for in-network caching of peer-to-peer content chunks: an analytical modelling study. In: 5th International Conference on the Network of the Future (NOF), 03-05 Dec 2014, Paris, France.

http://dx.doi.org/10.1109/NOF.2014.7119779

This is the Accepted Version of the final output.

UWL repository link: https://repository.uwl.ac.uk/id/eprint/2824/

Alternative formats: If you require this document in an alternative format, please contact: open.research@uwl.ac.uk

Copyright:

Copyright and moral rights for the publications made accessible in the public portal are retained by the authors and/or other copyright owners and it is a condition of accessing publications that users recognise and abide by the legal requirements associated with these rights.

Take down policy: If you believe that this document breaches copyright, please contact us at open.research@uwl.ac.uk providing details, and we will remove access to the work immediately and investigate your claim.

Rights Retention Statement:

Decision-Making for In-Network Caching of Peer-to-Peer Content Chunks - An Analytical Modelling Study

Xu Zhang, Ning Wang, Vassilios G. Vassilakis, Michael P. Howarth
Faculty of Engineering and Physical Sciences
University of Surrey, Guildford, Surrey, UK
Email: {xu.zhang,n.wang,v.vasilakis,m.howarth}@surrey.ac.uk

***Abstract*—In-network content caching has recently emerged in the context of Information-Centric Networking (ICN), which allows content objects to be cached at the content router side. In this paper, we specifically focus on in-network caching of Peer-to-Peer (P2P) content objects for improving both service and operation efficiencies. We propose an intelligent in-network caching scheme of P2P content chunks, aiming to reduce P2P-based content traffic load and also to achieve improved content distribution performance. Towards this end, the proposed holistic decision-making logic takes into account context information on the P2P characteristics such as chunk availability. In addition, we also analyse the benefit of coordination between neighbouring content routers when making caching decisions in order to avoid duplicated P2P chunk caching nearby. An analytical modelling framework is developed to quantitatively evaluate the efficiency of the proposed in-network caching scheme.**

I. Introduction

Information- or more specifically Content-Centric Networks (ICN/CCN) [1], [4] has been introduced in recent years to effectively deliver named content objects in the network rather than relying on the traditional host-to-host communication model. A key feature of ICN/CCN is that content objects can be cached within the network for local access by future interested clients. On the other hand, given the limited capacity of network devices, the management of content distribution among caches at the router side, or the *in-network caching* becomes crucial. Current in-network caching schemes mainly focus on generic client-server based paradigms, while how in-network caching operates for chunk based peer-to-peer (P2P) content has not yet been addressed in the literature. In particular, content chunks in P2P applications are delivered to client peers from a number of source peers in parallel, rather than from a single *always-on* server in the CDN scenario. Due to the fact that content chunks are contributed by multiple source peers rather than a single server, selfish behaviours of peers may lead to unexpected churns, resulting in unstable availabilities of demanded content objects that may deteriorate users' perceived service quality. As such, more "unselfish" in-network caching can be intuitively regarded as a useful approach to mitigate such adverse effects.

It is worth mentioning that in-network caching for chunk-based P2P content poses distinct technical challenges, and these have not yet been thoroughly addressed until now. Important factors for P2P chunk caching, such as the content chunk *availability* (due to peer join/leaving churns), need to be specifically considered. Other conventional context information, such as content *popularity*, also needs to be considered when making comprehensive P2P content caching/replacement decisions.

In this paper, we aim to holistically address the aforementioned research challenges related to in-network caching of chunk-based P2P content. The decision-making logic behind such caching operations takes into account important context information, with the objective of assuring caching efficiency and content distribution quality. We start from a simple scenario with *uncoordinated* decision-making by individual content routers (CRs) for caching passing-through P2P content chunks. Such a scenario takes into account metrics for enabling *locally* optimised caching decision, including chunk popularity/availability. Out of these parameters, chunk availability is specific for the P2P content distribution scenario. In order to evaluate the in-network caching policy, a dedicated mathematical model is developed to analyse such content management. In addition, we also consider locally *coordinated* caching in order to avoid unnecessarily duplicated caching between nearby routers. Our work also offers insights into provisioning strategies on content routers' caching resources for supporting caching of P2P content chunks in dynamic environments.

II. Related Work

In the recent IETF proposals, [2], [3] and [24] propose two options to cache content chunks within the network, either *implicitly* or *explicitly*. With implicit caching, content routers passively cache passing-through content chunks, while the explicit policy enables routers to explicitly request for content chunks to be cached locally. [2] explored in-network caching related to P2P systems. A random caching policy was adopted at each router, while the issue of what to cache is not addressed.

In recent research works in the context of ICN, [4],[5] promote a universal in-network caching strategy. However, high caching redundancy can be introduced as observed in [6], and thus the authors suggested a probabilistic caching algorithm for distribution of content caching along a content

delivery path, based on caching capacities of content routers for a given path per unit time. A simpler policy is proposed in [15], in which caching decisions are taken uniformly with a fixed probability. Similarly, a random autonomous caching scheme is adopted in [8], suggesting caching data packets and processing data requests in a random way, such that a simple load-sharing between routers can be provided. [27] develops a Markov model to evaluate the in-network caching performance, and they show the performance gain under universal caching compared to non-caching scenario.

In the context of coordinated in-network caching, there are only a few relevant works [9]-[14], [16][25]. [12]-[14] mainly explored the forwarding and routing scheme, while [16] focused on traffic engineering's point of view to promote collaboration in a two-level hierarchical manner, with the adoption of a universal caching policy. [10] and [11] both promoted implicit coordination to cache popular chunks at routers along a delivery path. And [9] proposed a collaboration scheme that goes beyond the en-route routers, the caching decision of which is based on solely content popularity measured in a wide scope of the whole network. Authors in [25] mainly explored the optimality of coordination among content routers of CCN, by developing an optimization model to deal with the trade-offs between network performance and the coordination cost.

[7], [18] and [19] are the only identified research works related to P2P in-network caching. Authors in [7] suggested to deploy in-network caches inside access networks rather than in every cacheable network device, in order to localize overlay traffic inside a network to avoid costly inter-domain traffic and to achieve the improved performance of end users. While [18] established a test platform to show the basic benefits of in-network caching in P2P systems in terms of traffic redundancy elimination, the content distribution at routers still followed a random way. And [19] mainly explored the cache placement policy along the delivery paths without considering caching decisions.

Different from the above research works, we focus on the development of efficient in-network caching policies of P2P contents in this paper, from both independent and coordinated perspectives. Specifically, due to dynamic P2P features, content availability has become a critical issue for P2P systems [20], and this is specifically taken into account in our work. Although [21] explored an optimal content placement, they mainly focused on the content caching at end peers in the scenario of video-on-demand P2P, wherein content popularity is solely taken into account. A fluid model is developed in [22] to minimize the inter-domain traffic for caches deployed at the boundary of ISPs, while how efficient caching policies can be achieved is not explored.

III. In-network Caching of P2P Content Chunks

A. Scheme overview

It is worth mentioning that, it is not our objective to provide detailed specification of in-network caching based on specific P2P applications. Instead we mainly focus on theoretical analysis on the *decision-making logic* in performing such operations in P2P environments. This work is mainly based on *implicit caching* of P2P content chunks, in which content routers do not actively request content chunks from peer users. According to the proposed caching scheme, content routers have the capability of caching passing-through P2P chunks according to specific policies. On the other hand, each content router may intercept P2P chunk availability advertisements originated from application source peers on the user side. An intermediate content router may update a passing-through availability advertisement provided that: (1) the router has locally cached content chunk(s) that have relevance to the chunks being advertised (e.g. belonging to the same content object like a movie); and (2) there are locally cached chunks that are not advertised in the original availability message from the source peer. If this is the case, the content router may insert in an opaque manner the identifier of the "missing" relevant chunks it has already cached into the advertisement message. Thereafter, the content router then forwards the updated advertisement to the targeted requesting peers. In this case, the requesting peers become aware of the chunk availability from the original source peer, and hence may send chunk requests towards it. When the intermediate content router receives such a request, it is able to serve the requesting peer with the chunks that have been locally cached.

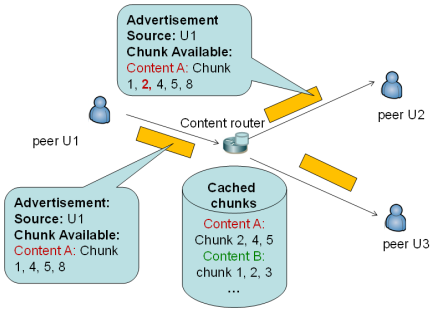

Fig. 1. An example for in-network P2P chunk caching

Fig. 1 illustrates a simple scenario on the basic operation of in-network P2P chunk caching. Assume that the intermediate content router has cached chunks No. 2, 4 and 5 for content item A. Peer $U1$ sends out its chunk availability advertisement for Content A (including chunks 1, 4, 5, 8). When the content router receives the message, it may silently insert chunk ID of 2 into the advertisement and forwards it to other potentially interested peers. As such, when peer $U2$ receives the message including chunks 1, 2, 4, 5 and 8, it assumes all of them can be provided by peer $U1$. When the intermediate router has received the content requests for those chunks, it is able to serve chunk 2 that has been locally cached. This type of in-network P2P caching and serving operation is transparent to the peers at the user side. It can be inferred that in order to support such an in-network chunk caching operation, routing needs to be symmetric within the network. Indeed this has been the common practice for both IP and MPLS networks within a single autonomous domain. How such a technique can

be applied at the Inter-domain scale (with possibly asymmetric BGP routing) will be left for our future study.

The chunk availability is a specific concern for P2P systems, since as noted in [20] 40% of the swarms suffer from the lack of available publishers for more than 50% of the time. As such, severe access burden to the source peer can be incurred. Therefore, content unavailability issue is critical for P2P systems. Let A_j denote chunk j's availability intensity, expressed as $\frac{a_j}{\sum_i a_i}$, given a_j as the number of replicas of chunk j available at source peers. Thus $(1 - A_j)$ indicates chunk j's unavailability degree. Let Pop_j denote chunk j's popularity degree, expressed as $\frac{n_{req_j}}{\sum_i req_i}$, given n_{req_j} as the number of requests for chunk j received at the content router. $\sum_i a_i$ and $\sum_i req_i$ are the normalized factor with respect to all the cached items in the router. Since in the ICN scenario [4], common requests for the same content object can be identified and thus be aggregated at a router, the value of n_{req_j} can thus be obtained. And the value of a_j can be obtained during the procedure of content advertisement originated from source peers when establishing connections with each other, as stated previously of Fig. 1.

Then the content routers are configured to be able to cache in-coming chunks in a probabilistic way as the following. For the rest of the paper, we use "router" and "cache" interchangeably to refer to the cacheable intermediate node.

$$Prob_{caching}(j) = Pop_j \cdot (1 - A_j) \tag{1}$$

Hence, based on the context information above, content routers can locally serve follow-up content chunk requests with cached chunks by *intercepting* the requests from end peers. However, such independent decision-making can still incur caching inefficiency, such as huge cache redundancy, due to the lack of awareness of other routers' caching states. Therefore, it is necessary to enable coordination between neighbouring routers so as to eliminate the cache redundancy *nearby*. In order to achieve this, neighbouring content routers may periodically exchange with each other a list of locally cached P2P chunks. For scalability consideration, neighboring routers may use Bloom Filter (BF) based techniques to aggregate the identifiers of their currently cached chunks before exchanging such information with each other. Such a technique has been applied in a recently work on ICN [23], and detailed description on its adaptation to our scheme will not be elaborated in this paper. With coordination, the router is able to check the latest information advertised by its neighbouring routers regarding their cached chunks. If the chunk has already been cached elsewhere, the router simply forwards it to downstream routers without caching it locally. Otherwise, the router makes caching decision based on Eq. (1).

B. In-network P2P Caching Model

In the following we develop a simple model to investigate the caching performance of in-network P2P caching. Following the common practice in relevant works, a unified chunk size is assumed and thus the cache capacity can be expressed as the number of chunks, defined as n. We assume requests for the chunk indexed as j arrive as a Poisson process with rate λ_j following the common assumption of [27]. A cached item can be evicted based on a Least Recently Used (LRU) or a Least Frequently Used (LFU) cache eviction policy. In some cases, a random eviction policy [17] is adopted due to its simplicity and line-speed caching operations.

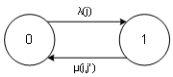

Fig. 2. CTMC of the caching process at a single router of a chunk

1) Model of the Caching Process of A Single Independent Router: The caching states of a chunk j can be modelled by a two-state continuous-time Markov chain (CTMC) in Fig. 2. State (0) corresponds to a non-cached case and state (1) corresponds to a cached-case of the chunk. The transition from state (0) to state (1) indicates a caching action of the chunk, with rate at $\lambda(j)$. And the transition from state (1) to state (0) implies an eviction action of the chunk, with rate at $\mu(j, j')$, where j' is the index of the following in-coming chunk when chunk j is cached. $\lambda(j)$ mainly depends on two factors: the caching policy ($Prob_{caching}(j)$) and the corresponding requests pending at the router. Requests arrival rate (λ_j) indicates the demand for the chunk recorded at a router, while the caching policy determines the possibility for the chunk to be cached. Based on the cache replacement policy, a cached item can be replaced by an in-coming chunk only if a caching decision is made when the cache is fully occupied. If a random eviction policy is applied, the probability of any cached item to be replaced is $1/n$, where n is the cache capacity. Under the LFU eviction policy, the probability of the cached item j to be evicted can be formulated to $\frac{1-Pop_j}{n}$. Then transition rates of $\lambda(j)$ and $\mu(j, j')$ can thus be defined as the following, respectively.

$$\lambda(j) = \lambda_j \cdot Prob_{caching}(j) \tag{2}$$

$$\mu(j, j') = \lambda(j') \cdot Prob_{eviction}(j) \tag{3}$$

where $Prob_{eviction}(j)$ is the probability that the cached chunk j is to be replaced by the in-coming chunk j' based on Eq. (1). Note $\lambda(j')$ can be an average value of all the in-coming chunks when the analysis is conducted.

Then according to the Markov property [26], we can derive the steady-state probability P_i that the system is in state (i), along with the boundary condition $\sum_i P_i = 1$, $i \in \{0, 1\}$. Thus the lifetime of the chunk j cached can be derived based on Eq. (2) and (3) as

$$P_1(j, j') = \frac{\lambda(j)}{\lambda(j) + \mu(j, j')} \tag{4}$$

A longer lifetime of a chunk in the cache implies that more requests received at the router can be satisfied, which is directly correlated to cache hits [27].

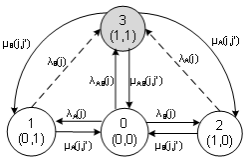

Fig. 3. CTMC of coordinated caching process among two neighbouring routers of the same content chunk

2) Model of the Caching Process of Multiple Coordinating Routers: In order to investigate the cache redundancy and the diversity of content distribution between neighbouring routers, we extend our analysis above to a multi-router modelling. The caching state (i) ($i \in \{0, 1, \ldots, 2^n - 1\}$) is characterized as the i_{th} case regarding the distribution of a single chunk in a neighbourhood of n caches. Thus for a pool of n neighbouring routers, the overall caching states for a single chunk can be characterized as 2^n states. Specially, with n neighbouring routers coordinating with each other, state $\underbrace{(i, j, k, \ldots)}_{n}$ corresponds to a case of the distribution of a chunk among n neighbouring routers, wherein binary values of $i, j, k, \ldots = 1$ indicates a cached case of the chunk, otherwise $i, j, k, \ldots = 0$. From the analytical aspect, the goal of the coordinated caching can be formulated to reducing the *cache redundancy ratio* (R) of the probability that a single chunk is held by more than one router in a neighbourhood, so as to improve the diversity of content distribution.

For illustration simplicity, a CTMC model with two neighbouring routers (e.g., router A and B) is given in Fig. 3 to depict the distribution of a single chunk among two neighbouring routers. The dotted transitions show the case of caching process under non-cooperation, wherein a duplicate caching can happen even if one of the neighbouring routers already holds a replica. A direct transition from state (0,0) to state (1,1) can happen if the same data is demanded by multiple requestors via different delivery paths, wherein $\lambda_{AB}(j) = \lambda_A(j) \cdot \lambda_B(j)$, $\mu_{AB}(j, j') = \mu_A(j, j') \cdot \mu_B(j, j')$ based on Eq. (2) and (3). Then the cache redundancy ratio is calculated as $R = P_3(t)$ in Fig. 3.

IV. Numerical Results

In this section, we show the caching performance under the proposed caching policy, compared with the universal caching and the popularity-based caching policies.

We evaluate the lifetime ($P_1(j, j')$) of a chunk cached under different caching policies in Fig. 4, with popularity degree as $Pop_j = 0.5$, and availability degree as $A_j = 0.1$. Setting $Pop_{j'} = 0.6$ and $A_{j'} = 0.8$ for the newly incoming chunk j', with $\frac{\lambda'_j}{\lambda_j} = 1.2$, the results in Fig. 4 show the caching performance of a rare chunk. As compared, the proposed strategized caching policy outperforms other caching policies greatly, while the universal caching with random eviction policy is the worst, due to its blind-eviction of cached items. Note the popularity-based caching policy, however, can perform even worse than the universal caching policy, which

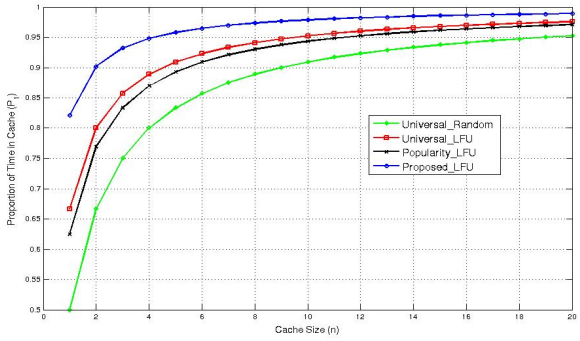

Fig. 4. Lifetime of the chunk cached in the router under different caching policies

is different from the CDN-based in-network caching [10]-[11] that claimed a better performance under the popularity-promoted caching. The rational is mainly due to P2P features that source peers are not always available as the always-on server of CDN, and thus for rare chunks considered in the analysis settings, the lifetime of them at a cache under the popularity-based caching policy can be small if its popularity is not ranked high. As such, a strategized caching policy with particular concern of content availabilities is crucial for dissemination of rare chunks in the network of P2P systems.

Next, we evaluate the impact of chunk availability on the caching performance. Following the settings above with cache size fixed, the lifetime of ($P_1(j, j')$) a chunk cached is plotted in Fig. 5, with its availability intensity varying from rare ($A_j = 0.1$) to ubiquitous at source peers ($A_j = 0.7$). As shown in the figure, distinct chunk can acquire the longest lifetime in a router under the proposed strategized caching policy, compared to other caching policies. The steady trend under native policies indicates the chunk availability-blind caching of in-coming chunks, while the decreasing trend under the proposed policy implies the promotion of caching rare chunks. As noted, the popularity-based caching policy can suffer from slightly worse performance than the universal_LFU caching policy. Similar reason as the above can be also applied here that only popular chunks are preferred to be cached under the popularity-based policy.

As such, based on results from Fig. 4 and 5, we observe that a popularity-based caching policy is not promising for in-network P2P caching, due to the chunk availability dynamics at source peers. On the other hand, with particular concern of context information, especially chunk availability, a more strategized caching policy can help to achieve efficient data distribution among caches in the network for P2P content chunks.

Next, we evaluate the cache redundancy between coordination and incoordination. The cache redundancy ratio based on the model in Fig. 3 is shown in Fig. 6. As shown in the figure, the trend of the cache redundancy is increasing with time elapsing due to Markov property of transient probabilistic results, with $t \to \infty$, the results reach steady states. Given the

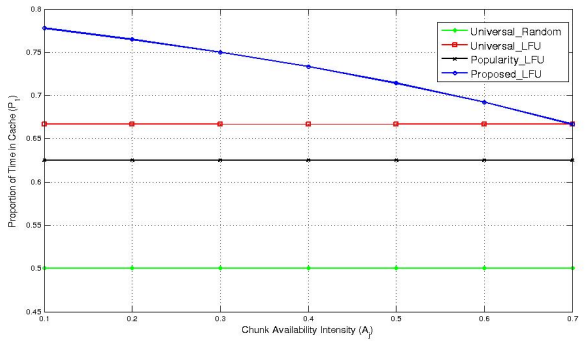

Fig. 5. Lifetime of the chunk cached in the router with chunk availability varying

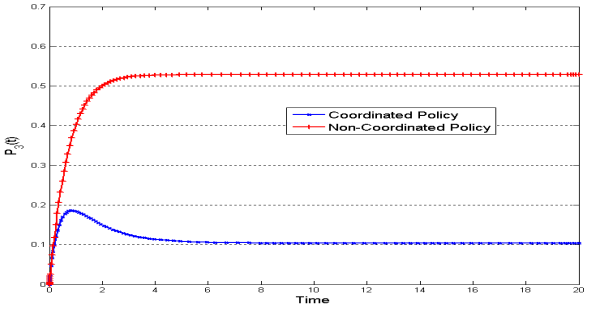

Fig. 6. Comparison of cache redundancy ratio under coordinated in-network caching and non-coordinated in-network caching based on two nearby routers

same value of $\lambda(j)/\mu(j,j') = 2.7$ for both routers, indicating high caching probability of the chunk j, the cache redundancy ratio can be reduced by 5 times as observed from the figure, from around 50% under the uncoordinated scenario down to around 10% with the coordinated scenario. It is worth mentioning that such a dramatic gain in caching redundancy reduction should be regarded as the theoretical upper bound. In practice, to use Bloom Filters to aggregate chunk identifiers may suffer from false positives, and the impact of such false positives to the overall caching performance will be evaluated in our future work.

V. Conclusion

In this paper, we proposed an efficient in-network caching decision-making of P2P content chunks, with both independent and coordinated caching concerned. A mathematical framework based on Markov Chain is developed in order to evaluate the caching performance for various caching policies. Concerning the avoidance of duplicated caching between nearby routers, exchange of necessary context information with adjacent routers is incorporated into caching decisions' making. Numerical results show that the proposed strategized caching policy can achieve desirable caching performance and at the same time, huge traffic redundancy as well as cache redundancy can be reduced, such that access load from source peers can be greatly alleviated.

Acknowledgment

This work is partially funded by the ChistEra CONCERT Project (A Context-Adaptive Content Ecosystem Under uncertainty).

References

[1] T. Koponen, et al., *A Data-Oriented (and beyond) Network Architecure*, SIGCOMM, vol. 37, no. 4, pp. 181-192, 2007.
[2] H. Song, et al., *DECoupled Application Data Enroute (DECADE) Problem Statement*, IETF RFC 6646, 2012.
[3] R. Alimi, et al., *A Survey of In-Network Storage Systems*, IETF RFC 6392, 2011.
[4] Van Jacobson, et al., *Networking Named Content*, Proc. of the 5th international conference on Emerging networking experiments and technologies, ACM CoNEXT'09, 2009.
[5] E. J. Rosensweig, et al., *Breadcrumbs: efficient, best-effort content location in cache networks*, in Proc. of INFOCOM, 2009.
[6] Ioannis Psaras, et al., *Probabilistic In-Network Caching for Information-Centric Networks*, in ACM SIGCOMM 2012.
[7] K. Katsaros, et al., *MultiCache: An overlay architecture for information-centric networking*, in Proc. of Computer Networks, vol. 55, no. 4, 2011.
[8] Somaya Arianfar, et al., *On Content-Centric Router Design and Implications*, in ACM ReArch 2010.
[9] Shuo Guo, et al., *Collaborative Forwarding and Caching in Content Centric Networks*, in ACM ReArch 2010.
[10] Kideok Cho, et al., *WAVE: Popularity-based and Collaborative In-network Caching for Content-Oriented Networks*, in Proc. of IEEE INFOCOM Workshop on Emerging Design Choices in Name-Oriented Networking, 2012.
[11] Zhongxing Ming, et al., *Age-based Cooperative Caching in Information-Centric Networks*, in Proc. of IEEE INFOCOM Workshop on Emerging Design Choices in Name-Oriented Networking, 2012.
[12] Lijun Dong, et al., *Optimal Caching with Content Broadcast in Cache-and-Forward Networks*, In Proc. of IEEE ICC, 2011.
[13] Walter Wong, et al., *Content Routers: Fetching Data on Network Path*, in Proc. of IEEE ICC, 2011.
[14] Walter Wong, et al., *Neighborhood Search and Admission Control in Cooperative Caching Networks*, in Proc. of IEEE Globecom, 2012.
[15] S. Arianfar, et al., *Packet-level Caching for Information-centric Networking*, in ACM SIGCOMM, ReArch Workshop, 2010.
[16] Haiyong Xie, et al., *TECC: Towards Collaborative In-network Caching Guided by Traffic Engineering*, In Proc. of IEEE INFOCOM, 2012.
[17] Dario Rossi, et al., *A Dive into the Caching Performance of Content Centric Networking*, in IEEE International Workshop on Computer Aided Modelling and Design of Communication Links and Networks (CAMAD), 2012.
[18] Shu Yamamoto, *In-Network P2P Packet Cache Processing using Scalable P2P Network Test Platform*, in IEEE P2P, 2011.
[19] Mingjiang Ye, et al., *Caching the P2P Traffic in ISP Network*, in Proc. of IEEE ICC, 2008.
[20] Daniel S., et al., *Content Availability and Bundling in Swarming Systems*, in IEEE/ACM Tran. on Networking, Vol. 21, No. 2, 2013.
[21] Bo Tan, et al., *Optimal Content Placement for Peer-to-Peer Video-on-Demand Systems*, in IEEE/ACM Tran. on Networking, Vol. 21, No. 2, 2013.
[22] Frank Lehrieder, et al., *Caching for BitTorrent-Like P2P Systems: A Simple Fluid Model and Its Implications*, in IEEE/ACM Tran. on Networking, Vol. 20, No. 4, 2012.
[23] H. Liu, et al., *A Multi-Level DHT Routing Framework with Aggregation*, in ACM SIGCOMM ICN'12, Helsinki, Finland, August 2012.
[24] H. Song, et al., *DECoupled Application Data Enroute (DECADE)*, IETF RFC 7069, 2013.
[25] Yanhua Li, et al, *Coordinating In-network Caching in Content-Centric Networks: Model and Analysis*, IEEE International Conference on Distributed Computing System (ICDCS), 2013.
[26] Boudewijn R. Haverkort, *Performance of Computer Communication Systems*, Wiley, 1998.
[27] Ioannis Psaras, et al., *Modelling and Evaluation of CCN-Caching Trees*, IFIP, Networking, 2011.