Joint Optimization of Caching and Routing Strategies in Content Delivery Networks: A Big Data Case

Xianchen Guo*, Tianyu Wang*†, and Shaowei Wang*
*School of Electronic Science and Engineering, Nanjing University, Nanjing 210023, China
†National Mobile Communications Research Laboratory, Southeast University, Nanjing 210096, China
Email: MF1723018@smail.nju.edu.cn, {tianyu.alex.wang, wangsw}@nju.edu.cn

***Abstract*—Content delivery networks (CDNs) have been proposed to improve the performance of large-scale content delivery in communication networks, in which content files are dynamically cached in CDN nodes that are close to end-users so as to decrease the transmission latency and traffic redundancy. We investigate a real CDN that is currently utilized by a large social network company in China. We note that there are two important issues that are not well considered in the existing literature. The first is that end-users can usually access multiple CDN nodes with high quality of experience. The second is that the data service price may vary greatly for different regions, which can highly influence the cost of CDNs for large-scale applications. We reconsider the optimal caching and routing problem in CDNs by jointly considering these two practical issues, and propose a joint caching and routing strategy by using the alternating optimization technique. Simulation results show that the proposed algorithm outperforms the current CDN strategy, in which most popular files are cached and end-users are directed to the nearest CDN nodes, by 30% and 12% in terms of latency and data service cost, respectively.**

I. Introduction

Due to the rapid development of wireless communication networks, especially 4G, wireless traffic has experienced an explosive growth in the past few years. According to the report of Cisco, by 2019, video traffic, including TV, Internet, video on demand (VoD), and peer-to-peer (P2P), will constitute approximately 80 to 90 percent of global consumer traffic, and 70 percent of the IP VoD will be high-definition video. In terms of access modes, mobile data traffic will increase 10 fold from 2014 to 2019, and video traffic will account for 72 percent of global mobile data traffic [1]. In order to cope with such dramatic increase of wireless traffic, the traditional content delivery networks (CDNs) have draw a lot of attention in the domain of wireless applications. CDNs utilize the caching capability of servers that are close to end-users to build a virtual network to facilitate the content delivery process, in which popular contents are cached in multiple CDN servers to reduce redundant transmission and fetching delay [2]. In CDNs, a variety of factors can highly influence the network performance, including traffic status, network connectivity, demand distribution and content popularity, which have drawn a lot of research topics in the literature [3].

Cache management including determining the location and size of each cache, selecting which file should be cached in which node, and how to transfer content to these caches is one of the most important issues in CDNs [4]–[6]. In order to optimize the network performance, the content files that are cached in the CDN servers should be carefully managed, such that the probability that the requested files can be found in the server is maximized. To increase the hit rate of CDN servers, the popularity of different files in different time period and different locations should be accurately predicted. The popular contents not only include the total content popularity in local areas, but also have individual user preference [7]. Investigations show us only a portion of content files contribute to most of the data traffic due to frequent and repeated downloads [8]. Therefore, if we can predict file popularity to intelligently formulate content placement strategies, we will be able to reduce expenses of caching files that are accessed only once and duplicate content transmission costs. For example, a popular caching strategy is proposed in [9], where the content popularity is exploited to determine which content should be stored.However, the well-known and widely-used content placement policies including the first-in first-out (FIFO) scheme, the least recently used (LRU) scheme and the least frequently used (LFU) scheme, ignore the importance of popularity. In addition, there are a variety of factors that synthetically affect the overall caching performance, including the capacity of each cache in CDNs, the content popularity during different time periods, the location of users and the network system framework. [10] and [11] have focused on the cache management problem, i.e., determining the placement of each file for a given topology and popularity distribution while [12] and [13] have researched a joint cache management and routing problem, considering the inter-related routing and caching decision.

In this paper, we consider a real CDN deployed by a top Internet business in China, in which millions of people share their life moments by pictures and videos. We note that some important issues are highly concerned by the CDN operators but are not well studied in the literature:

This work was partially supported by the National Natural Science Foundation of China (61671233,61801208), the Jiangsu Science Foundation (BK20170650), the Postdoctoral Science Foundation of China (BX201700118,2017M621712), the Jiangsu Postdoctoral Science Foundation(1701118B), and the open research fund of National Mobile Communications Research Laboratory (2019D02).

978-1-5386-8088-9/19/$31.00 ©2019 IEEE

- The popularity of social contents can highly deviate from the classic Zipf distribution. Users in regions show different interests and they may change over time. From our investigation, about 60 percent of files are requested only once and files requested more than 4 times can be regarded as popular contents.
- The prices of data service can dramatically change from region to region. For example, unit price in Tianjin is 32.8 and that in Shanghai is 33.17, so these big cities usually contribute to a large portion of content traffic, which highly increase the network operational cost.
- Mobile users can be directed to nearby CDN servers and still have good performance. As the development of CDNs, the number of nodes used to serve users in the same place increases. Across the country, the CDN operator has about 50 different CDN nodes providing services to users.

In this paper, we reconsider the caching and routing strategies of CDNs, in which the contents popularity, prices, are joint considered. In order to maximize quality of experience and decrease costs of operator, we propose a joint cache management and routing algorithm by using alternating optimization. Simulation results show that our algorithm yield significant performance improvement contrast with two content placement policies, most popular caching placement policy and random caching placement policy.

The rest of this paper is organized as follows. In Section II, we provide the system model and problem formulation. . In Section III, we analyze the optimization problem and propose a joint optimization method using alternating optimization. In Section IV, we provide the simulation results. In Section V, we draw our conclusion.

II. System Model and Problem Formulation

A. System Model

We built the system model according to an authentic network architecture followed by Fig.1. We view users within a region as a whole and give I regions covered by a CDN with K caches. Z_i is denoted as the number of users in region i. The sets of users and caches are denoted as $\mathcal{I} = \{1, 2, \ldots, I\}$ and $\mathcal{K} = \{1, 2, \ldots, K\}$ respectively and we assume that $K \leq I$. Caches are connected to the zone/back-end server and the upper limit of the number of the access user for each caches is denoted by A_k. There are J files to be cached, which are denoted as $\mathcal{J} = \{1, 2, \ldots, J\}$. The size of each file is uniform and the cache capacity at each CDN is denoted as C_k. We denote the probability that user i requests file j by $q_{i,j}$,i.e.,the popularity of file j. λ_i denotes as a Poisson process of aggregate rate, i.e., the number of user I requesting for files per unit time while λ is denoted as mean value of this process.

Therefore, in our model, I users request for J files from K caches. User can receive services from all caches theoretically suffered from different delays. When the file requested by the user cannot be queried in the cache, cache can ask the zone to push the corresponding file to serve the user. This process is referred to as cache miss. On the contrary, cache can directly and effectively serves the user when the file requested is queried in it, which is called cache hit. Obviously, cost and delay of cache miss are both greater than cache hit. We denote average delay of user I accessing cache K in the case of cache hit or miss by $\gamma_{i,k}$ and $\theta_{i,k}$. Meanwhile, the cost of cache k in the event of cache hit or miss is denoted by α_k and β_k. We suppose that $\alpha_k \leq \beta_k$ and $\gamma_{i,k} \leq \theta_{i,k}$.

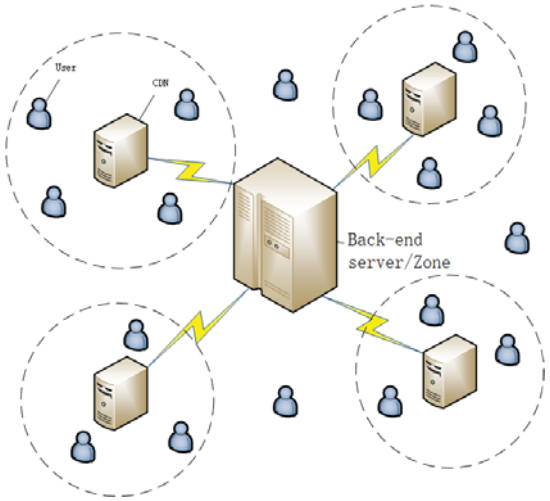

Fig. 1. System Model

B. Problem Formulation

We consider a joint content placement and routing problem with the goal of minimizing both cost and latency over all user requests for all files. Therefore, we should optimize both content placement and routing policies. For our content placement policy, we define a binary variable $X_{j,k} \in \{0, 1\}$ which denotes whether file j has been stored in cache k. If stored, $X_{j,k} = 1$ and otherwise $X_{j,k} = 0$. What is more, for our routing policies, we also define a variable $P_{i,k}$ that denotes the proportion of user i requests for files from cache k in user i from all caches.

We designed an optimization in which the expected cost and delay obtained by a content placement strategy $X = [X_{j,k}]$ and a routing strategy$P = [P_{i,k}]$ are respectively denoted by $Q(X, P)$ and $D(X, P)$. The goal of the joint content placement and routing problem is to minimize cost

$$Q(X,P) = \sum_{i\in\mathcal{I},j\in\mathcal{J}} \lambda_i q_{i,j} \Big[\sum_{k\in\mathcal{K}} P_{i,k} q_{i,j} X_{j,k} \alpha_k + \sum_{k\in\mathcal{K}} P_{i,k} q_{i,j} (1 - X_{j,k}) \beta_k\Big] \quad (1)$$

and latency

$$D(X,P) = \sum_{i\in\mathcal{I},j\in\mathcal{J}} \lambda_i q_{i,j} \Big[\sum_{k\in\mathcal{K}} P_{i,k} q_{i,j} X_{j,k} \gamma_k + \sum_{k\in\mathcal{K}} P_{i,k} q_{i,j} (1 - X_{j,k}) \theta_k\Big] \quad (2)$$

at the same time. Where $P_{i,k}q_{i,j}$ denotes as the probability that user I requests file J from cache K. Therefore, $P_{i,k}q_{i,j}X_{j,k}$ and $P_{i,k}q_{i,j}(1-X_{j,k})$ denote as the process of cache hit and cache miss respectively.

We aim to minimize the total latency and cost over this network while some practical constraints are taken into consideration including the storage capacity of caches in CDNs, the maximum number of users that each CDN allows. Mathematically, the joint content placement and routing optimization task ($OP1$) is as follows:

$$\begin{aligned} OP1 \quad & \min_{X,P} Q(X,P) + \sigma D(X,P) \\ s.t. \quad & C_1 : \textstyle\sum_{j\in\mathcal{J}} X_{j,k} \le C_k, \forall k \in \mathcal{K} \\ & C_2 : X_{j,k} \in \{0,1\}, \forall j \in \mathcal{J}, k \in \mathcal{K} \\ & C_3 : \textstyle\sum_{k\in\mathcal{K}} P_{i,k} = 1, \forall i \in \mathcal{I} \\ & C_4 : \textstyle\sum\limits_{i\in\mathcal{I}} Z_i P_{i,k} \le A_k, \forall k \in \mathcal{K} \\ & C_5 : 0 \le P_{i,k} \le 1, \forall i \in \mathcal{I}, k \in \mathcal{K}, \end{aligned} \tag{3}$$

in which σ denotes as the weight between cost and latency. $X_{j,k} = 1$ indicates file j is cached at cache k, otherwise, $X_{j,k} = 0$. C_k is the capacity of CDN k, so C_1 represents that the total size of files cached in each CDN node does not exceed its inherent capacity while C_4 shows the restriction on communication resources for each CDN node. C_3 indicates that the summary of probability of each user accessing each CDN node is 1.

III. Alternating Optimization Algorithm

A. Problem Decomposition

$OP1$ is a Mixed-Integer Program which is hard to solve. Because $OP1$ is a joint optimization of content placement and routing strategies for latency and cost minimization, it is quite natural to divide the optimization into two parts, respective minimization of cost ($OP2$) and latency ($OP3$) [14]

$$\begin{aligned} OP2 \quad & \min_X Q(X,P) \\ s.t. \quad & C_1 : \textstyle\sum_{j\in\mathcal{J}} X_{j,k} \le C_k, \forall k \in \mathcal{K} \\ & C_2 : X_{j,k} \in \{0,1\}, \forall j \in \mathcal{J}, k \in \mathcal{K} \end{aligned} \tag{4}$$

and

$$\begin{aligned} OP3 \quad & \min_P D(X,P) \\ s.t. \quad & C_3 : \textstyle\sum_{k\in\mathcal{K}} P_{i,k} = 1, \forall i \in \mathcal{I} \\ & C_4 : \textstyle\sum\limits_{i\in\mathcal{I}} Z_i P_{i,k} \le A_k, \forall k \in \mathcal{K} \\ & C_5 : 0 \le P_{i,k} \le 1, \forall i \in \mathcal{I}, k \in \mathcal{K}, \end{aligned} \tag{5}$$

in which $OP2$ indicates the total cost determined by variable X while $OP3$ indicates the total delay for all users determined by variable P.

B. Algorithm of $OP2$ and $OP3$

Though observing $OP2$ and $OP3$, we should find that $OP2$ is an integer programming while $OP3$ is a convex programming. Because the objective function $D(X,P)$ is a linear function, the constraint of equality C_3 is affine and the constraint of inequality C_4 is convex. We want to obtain the linear relaxation of this integer programming by setting the domain of the variable X to be $0 \le X_{j,k} \le 1, \forall j \in \mathcal{J}, k \in \mathcal{K}$ and $OP2$ becomes $OP4$:

$$\begin{aligned} OP4 \quad & \min_X Q(X,P) \\ s.t. \quad & C_1 : \textstyle\sum_{j\in\mathcal{J}} X_{j,k} \le C_k, \forall k \in \mathcal{K} \\ & C_6 : 0 \le X_{j,k} \le 1, \forall j \in \mathcal{J}, k \in \mathcal{K}. \end{aligned} \tag{6}$$

We can use standard optimization techniques to solve $OP3$ and $OP4$ since these two problems are convex.

C. Alternating Optimization

From the above analysis, we are able to apply an algorithm of alternating optimization to solve the two problems. Because both $X_{j,k}$ and $P_{i,k}$ are in the two optimization objectives, we can fix one and update another, then iterate till convergence. Therefore, we first fix the content placement policy X and make all $X_{j,k} = 0$, i.e., all files are not stored in any cache. Next, we substitute X into $OP3$ to update routing policy P. Afterwards, we exchange fixed variable and substitute P just figured out into the $OP4$ and update content placement policy X in which elements have been rounded off to be 0 or 1. This is what we do at each iteration and we stop the iteration and output $Q(X,P)$ and $D(X,P)$ when they are no longer changing or the maximum number of iterations is reached. The specific process has been described by Algorithm 1.

Algorithm 1 Alternating Optimization

1: Initialize matrix X as a $j \times k$ null matrix
2: Initialize $\mathcal{N}$ as the maximum number of iteration
3: Initialize n = 0
4: **while** $n \le \mathcal{N}$ **do**
5: Substitute X into $OP3$ to update P
6: Substitute P into $OP4$ to update X
7: $i = i + 1$
8: **end while**

Output: Content placement policy X,routing policy P,summary of cost $Q(X)$,summary of delay $D(P)$.

IV. Simulation

In this section, we evaluate the performance of our alternating optimization algorithm through simulations. Our system framework is based on a real network system, and some data in our simulation is real, such as system classification and detailed functions, the unit price of each CDN node, the capacity of each CDN node and so on. Therefore, our simulation is of certain practical significance. According to regulations of the existing network system, each CDN node is responsible for all users in a region, that is to say, the latency in accessing the cache by users in the region where the cache is responsible is the least among all CDN nodes. The number of caches is not equal to the number of divided user areas, more

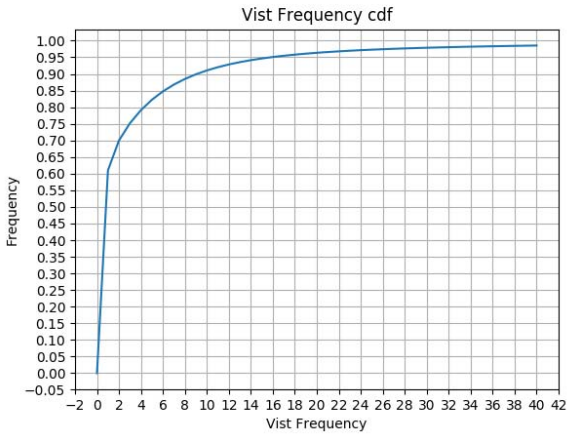

Fig. 2. cumulative distribution function describing popularity.

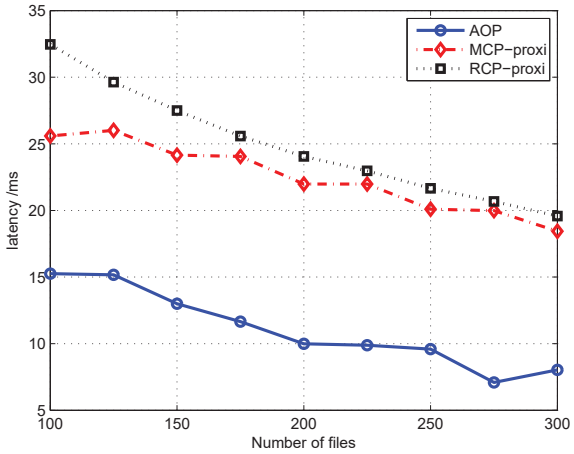

Fig. 3. Latency that users access CDNs as the number of files increases. The unit latency of congestion is 1ms.

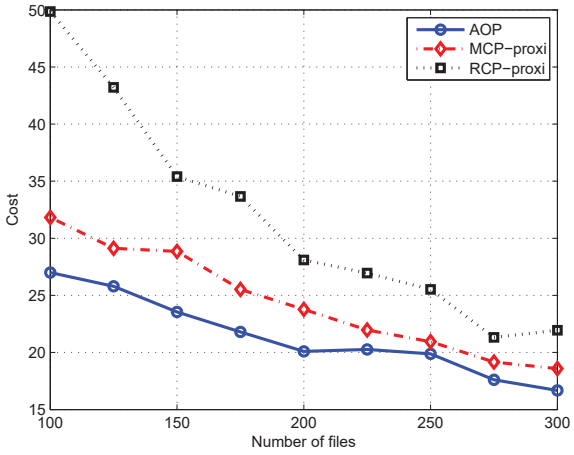

Fig. 4. Cost that users request for files as the number of files increases. The unit latency of congestion is 1ms.

exactly, based on real data k (the number of CDN nodes)$= 4$ and i (the number of divided user areas)$= 10$, i.e., there are some areas where users do not have their own CDN nodes, and users in these areas need to request for contents from a neighboring CDN node. Fig.2 shows a cumulative distribution function which describes the frequency accessed of files. Our practical investigations show about 60 percent of files are accessed only once and about 80 percent of files are accessed 4 times and less than 4 times, i.e., files accessed more than 4 times can be called popular files which contributes to the vast majority of data traffic. Due to our investigations, we denote the popularity of files requested by users as a Zipf distribution with the skewness parameter δ about 0.9 and λ as 5. As mentioned above, we denote the capacities of the four cache as $C = [77, 83, 124, 31]$ and the unit prices of the four are set as $\alpha = [32.8, 33.17, 28.4, 36.85]$ and $\beta = 3\alpha$. We have users in 10 regions and the request latency $\theta_{i,k}$ in the event of cache hit equals one-tenth of the distance between user$i$ and cache k while in the event of cache miss, the latency $\sigma_{i,k} = \theta_{i,k} + 25ms$ [12]. We don't have real data on A_k and z_i, but we set $\sum_i Z_i P^0_{i,k} > A_k$ for some $i \in \mathcal{I}$ in which $P^0_{i,k}$ is the routing policy determined by the principle of proximity. Otherwise, the routing policy is confirmed according to the principle of proximity invariably, i.e., the routing policy determined by principle of proximity has been always optimal.

We compare our proposed alternating optimization policy (AOP) with other representative content placement policies, most popular caching placement policy (MCP) and random caching placement policy (RCP) respectively. In detail, due to the distribution of popularity, we set each cache stores the most popular files while it still has spare capacity for MCP. For RCP, we set that each cache stores files randomly as long as its capacity is not exceeded. The routing policy we set for MCP and RCP is in terms of the the principle of proximity. We denote MCP-proxi and RCP-proxi as the two joint content placement and routing policies. As mentioned above, for some $i \in \mathcal{I}$, $\sum_i Z_i P^0_{i,k} > A_k$, so we set latency for congestion in which the unit number of users in congestion will bring about an additional latency of $1ms$, which is not the real data. According to the actual unit cost in different areas, we calculate the unit cost in different places in the simulation according to the proportion Fig. 3 and Fig. 4 show us the summary of latency and cost decresing for AOP, MCP-proxi and RCP-proxi when the unit latency of congestion is 1ms as the number of files increases. We can see that the performance of AOP is better than MCP-proxi while RCP-proxi's is always the worst. We adjust the unit latency of congestion and Fig. 5 shows us the summary of latency when the number of files is 150 and the number of users is 10 as the unit latency of congestion increase which indicates the relationship between the two kinds of latency. While congestion delay is little, transmission and fetching delay is predominant that cause

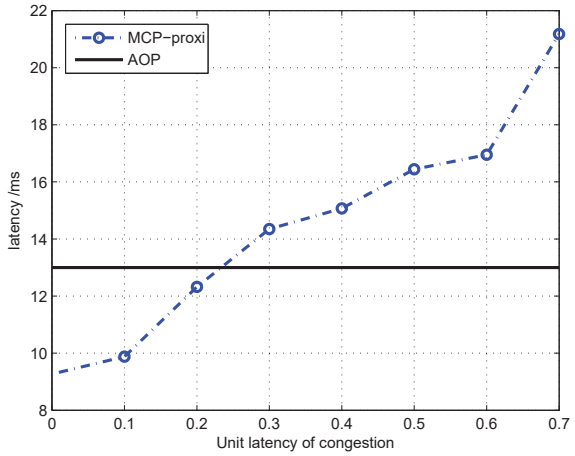

Fig. 5. Latency that users access CDNs as the unit latency of congestion increase. The number of files is 150.

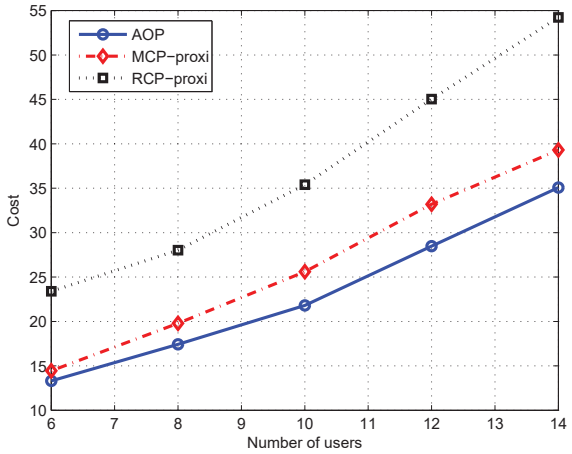

Fig. 7. Cost that users request for files as the number of users increases. The unit latency of congestion is 1ms and the number of files is 150.

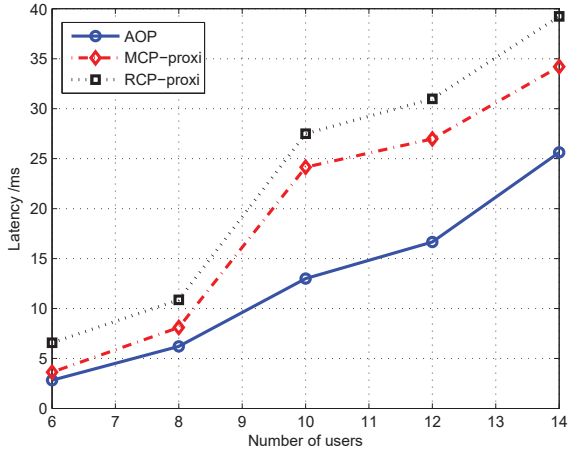

Fig. 6. Latency that users access CDNs as the number of users increases. The unit latency of congestion is 1ms and the number of files is 150.

the better performance of MCP-proxi. While the influence of congestion delay becomes larger, dynamic access routing is greatly essential. We also simulate the latency and cost under the change of the number of users, as shown in Fig. 6 and Fig. 7. We add 4 cities while the unit latency of congestion is 1ms and the number of files is 150. As the number of users increases, operator of the network have to take on more expenses. Meanwhile, the performance of our algorithm has been always the best, followed by MCP-proxi and RCP-proxi orderly.

V. Conclusion

In this paper, we have formulated the problem of joint content placement and routing policies in networks with a back-end server and CDNs. We build a network architecture that we consider it is universal according to the existing CDN network system. We considered minimizing the average latency and cost of users under the dual limitation of cache capacity and communication capacity. We reduce such a joint problem to a mixed-integer Program and then relax it to a convex problem. We developed an algorithm of alternating optimization which produces good results for this problem. Based on partially true data, our algorithm can converge very quickly and effectively with a great improvement compared with MCP and RCP revealed by the simulation results. Our future work is aimed at a more detailed theoretical study of this alternating optimization algorithm and developing distributed algorithms for content placement and delivery. Meanwhile, the popularity issue of videos should also be analyzed.

References

[1] J. Tang and T. Q. S. Quek, "The role of cloud computing in content centric mobile networking," *IEEE Commun. Mag.*, vol. 54, pp. 52-59, Aug. 2016.

[2] H. Liu, Z. Chen, and L. Qian, "The three primary colors of mobile systems," *IEEE Commun. Mag.*, vol. 54, pp. 15-21, Sep. 2016.

[3] R. Wang, X. Peng, J. Zhang, and K. B. Letaief, "Mobility-aware caching for content-centric wireless networks: modeling and methodology," *IEEE Commun. Mag.*, vol. 54, pp. 77-83, Aug. 2016.

[4] G. Paschos, E. Bastug, I. Land, G. Caire, M. Debbah, "Wireless caching: technical misconceptions and business barriers," *IEEE Commun. Mag.*, vol. 54, pp. 16-22, Aug. 2016.

[5] E. Zeydan *el al.*, "Big data caching for networking: moving from cloud to edge," *IEEE Commun. Mag.*, vol. 54, pp. 36-42, Sep. 2016.

[6] D. Liu, B. Chen, C. Yang, and A. F. Molisch, "Caching at the wireless edge: design aspects, challenges, and future directions," *IEEE Commun. Mag.*, vol. 54, pp. 22-28, Sep. 2016.

[7] M. A. Maddah-Ali and U. Niesen, "Coding for caching: fundamental limits and practical challenges," *IEEE Commun. Mag.*, vol. 54, pp. 23-29, Aug. 2016.

[8] S. Li, J. Xu, M. van der Schaar, and W. Li, "Popularity-driven content caching," *Proc. IEEE INFOCOM'16*, San Francisco, CA, USA, Apr. 2016.

[9] B. Ahlgren, C. Dannewitz, C. Imbrenda, D. Kutscher, and B. Ohlman, "A survey of information-centric networking," *IEEE Commun. Mag.*, vol. 50, pp. 26-36, Jul. 2012.

[10] I. Baev, R. Rajaraman, and C. Swamy, "Approximation Algorithms for Data Placement Problems," *SIAM J. Computing*, vol. 38, No. 4, pp. 1411-1429, 2008.

[11] M. Dehghan, B. Jiang, A. Seetharam, T. He, T. Salonidis, J. Kurose, D. Towsley, and R. Sitaraman, "On the complexity of optimal request routing and content caching in heterogeneous cache networks,"*IEEE/ACM Trans. Netw.*, vol. 25, pp. 1635-1648, Jun. 2017.

[12] P. Nuggehalli, V. Srinivasan, C. . Chiasserini, and R. R. Rao, "Efficient cache placement in multi-hop wireless networks," *IEEE/ACM Trans. Netw.*, vol. 14, pp. 1045-1055, Oct. 2006.

[13] A. Khreishah, J. Chakareski, and A. Gharaibeh, "Joint caching, routing, and channel assignment for collaborative small-cell cellular networks," *IEEE J. Sel. Areas Commun.*, vol. 34, pp. 2275-2284, Aug. 2016.

[14] J. C. Bezdek and R. J. Hathaway, "Some notes on alternating optimization," *LNCS*, vol. 2275, no. 4, pp. 288-300, 2002.