Cooperative Caching for Scalable Video Transmissions Over Heterogeneous Networks

Ticao Zhang and Shiwen Mao, *Fellow, IEEE*

***Abstract*—We investigate a cooperative video caching problem in heterogeneous networks (HetNet). In cellular networks with limited backhaul capacity, by equipping the small-cell base stations (SBSs) with caches, more data can be offloaded. The backhaul data is reduced and user equipment (UE) can enjoy a low latency. We formulate the total transmission delay (TTD) minimization problem as a nonlinear integer programming. To solve the NP-hard problem, we relax the constraints and propose a greedy algorithm for a feasible solution. We prove that the greedy algorithm is asymptotically optimal. Simulation results demonstrate that the proposed cooperative caching algorithm can significantly reduce the TTD.**

***Index Terms*—Cooperative caching, heterogeneous network, integer programming, scalable video transmission.**

I. Introduction

THE MOBILE data traffic is increasing rapidly and it is expected that about 78% of the total mobile traffic will be video related by the year 2021 [1]. Meanwhile, emerging video related applications even put more stringent requirements on both the throughput and latency of the mobile network. For example, a majority of video streaming is now becoming 1080p high resolution (HD), which incurs huge traffic flows. The virtual reality (VR) video streaming requires a throughput of 200Mbps and a latency as low as 10ms. The self-driving cars would not be able to work until we have 5G networks which has a super low latency to guarantee safety.

A promising approach to meet the unprecedented traffic demands and latency requirements in 5G wireless is mobile edge caching [2], where each BS can proactively store data (e.g., popular video files) in the off-peak hours at their storage devices, so that the heavy traffic burden in the backhaul can be effectively alleviated, especially when we consider the fact that a great deal of mobile data traffic is caused by repeated deliveries of popular video contents. Moreover, edge caching can significantly reduce the network latency due to the nature of store-and-forward design, which is of fundamental importance to latency sensitive applications such as VR.

Scalable video transmissions is a promising technology in heterogeneous networks (HetNet). In scalable video coding (SVC) [3], each video is encoded into one base layer (BL) and several enhancement layers (ELs). Higher video quality can be achieved by receiving more layers. This way, the dynamically varying wireless link bandwidth can be fully utilized based on user's different perceptual experiences on different kinds of video files. In HetNet, a macro base station (MBS) is deployed to provide a wide coverage of users in the macro cell, while small cell base stations (SBSs) located in the macro cell enable high data rates for subscriber UEs. This HetNet structure helps to improve system spectrum efficiency.

However, there are several challenges in this HetNet edge caching architecture for scalable videos. Compared with wired caching strategy in traditional content distribution networks (CDN) [4], the cache capacity of the SBSs is relatively small. How to cache different layers of the popular video files remains a challenging problem. In [5], the author proposed a heuristic approach to find an optimal caching placement strategy so that the average download time is minimized. However, the proposed solution is not efficient due to the assumption of non-cooperation among the BSs. In [6], a near optimal video layer placement strategy is proposed in a cache-enabled HetNet with stochastic geometry. However, in this letter, the authors assume that each video file layer can be cached at no more than one SBS. Considering that the most popular video files may be requested by users for many times, just storing one copy would be obviously not optimal.

Compared with the existing works, in this letter, we leverage both network topology information and video file popularity. We propose a cooperative framework in which different SBSs can cooperate with each other to jointly minimize the transmission delay of layered video files. The contributions of this letter are summarized as follows.

- We consider a cooperative caching scenario in which different SBS can share their local stored video files so that the total transmission delay (TTD) is reduced. Compared with noncooperative caching, cooperative caching is more general, effcient and practical.
- We prove the formulated problem is NP-hard, and propose a greedy algorithm that is asymptotically optimal as the size of the problem is increased, for video file placement in terms of reducing the transmission delay.
- The proposed cooperative caching algorithm can significantly reduce the TTD compared with the noncooperative caching and preference based caching. Besides, the impact of video file popularity, video caching size, and the number of BS are investigated. The simulation results validate the accuracy of our analysis and the superior performance of the proposed scheme.

Manuscript received March 25, 2019; revised April 9, 2019; accepted April 15, 2019. Date of publication April 18, 2019; date of current version May 23, 2019. This work was supported in part by the U.S. National Science Foundation under Grant CNS-1702957. The associate editor coordinating the review of this paper and approving it for publication was A. Ksentini. *(Corresponding author: Shiwen Mao).*

The authors are with the Department of Electrical and Computer Engineering, Auburn University, Auburn, AL 36849 USA (e-mail: tzz0031@tigermail.auburn.edu; smao@ieee.org).

Digital Object Identifier 10.1109/LNET.2019.2911972

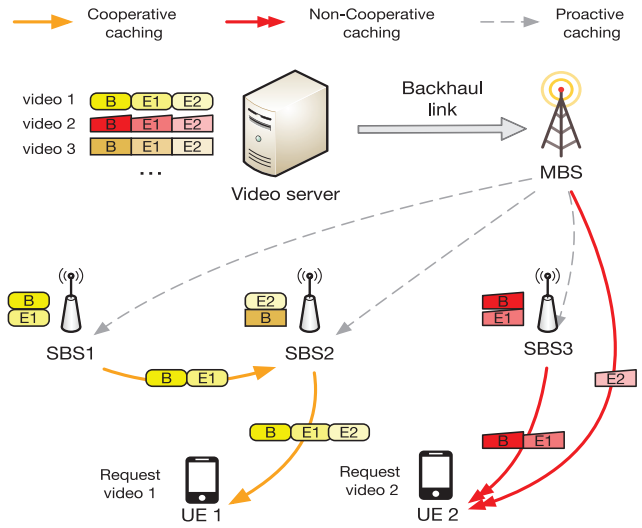

Fig. 1. Illustration of the system model and the cooperative caching scheme.

We introduce the system model in Section II and formulate the TTD minimization problem in Section III. We relax the problem into a convex one and propose a greedy algorithm in Section IV. Simulation results are presented in Section V. We conclude this letter in Section VI.

II. System Model

A. Network Topology

We consider a HetNet where a macro base station (MBS) is located in the center of the area, and B small cell base stations (SBSs) distribute uniformly in the area, as shown in Fig. 1. We denote the set of SBSs by $\mathcal{B}$ and index the MBS as $b = 0$, then all the potential BSs can be denoted as $\mathcal{B}^* = \mathcal{B} \cup \{0\}$. The network delay is defined as a symmetric matrix $\mathbf{D} \in \mathbb{R}^{(B+1)\times(B+1)}$ with the element $D_{a,b}$ denoting the transmission delay between BS a and BS b, for all $a, b \in \mathcal{B}^*$. Additionally, the delay from one BS to itself is zero, i.e., $D_{a,a} = 0$, for all $a \in \mathcal{B}^*$. The delay between different BSs can be measured and estimated using various existing methods [7], which is beyond the scope of this letter. Hence we always assume that each BS has a full knowledge of the delay matrix. Finally, compared to the delay between different SBSs, the delay between SBS and MBS is usually quite large.

B. SVC-Based Layer Caching

Scalable video transmission enables different perceptual experiences for different users. In SVC, each video file is divided into multiple layers, where the base layer (BL) provides a basic viewing quality and the enhancement layers (ELs) enable increasingly improved video quality. Typically, the decoding of an EL relies on the successful decoding of the BL and all the lower ELs. More decoded ELs generally lead to a more enhanced video quality.

Suppose there are L layers for each video: $l = 1$ represents the BL and $l = 2, 3, \ldots, L$ denote the ELs. For ease of analysis, we also assume all videos have the same number of layers and the size of each layer is set to 1 (i.e., a normalized file size). Suppose there are in all F videos in the video library $\mathcal{F}$. The videos are arranged in a descending order of popularity. Videos with smaller indices are considered to be more popular. The request popularity of videos follows the Zipf distribution with parameter γ as [8]

$$p(f) = f^{-\gamma} / \sum_{l=1}^{F} l^{-\gamma}, \quad f = 1, 2, \ldots, F, \tag{1}$$

where γ is the skewness parameter that characterizes the video popularity. Usually, a larger γ indicates a higher degree of request, i.e., popular video files account for the major requests.

Moreover, the users may have different preferences for video qualities. According to [9], the preference for the standard video quality (videos with only BL) is given by $g_{\mathrm{SDV}}(f) = \frac{f-1}{F-1}$ and the preference for the high definition video (videos with ELs) is $g_{\mathrm{HDV}}(f) = 1 - g_{\mathrm{SDV}}(f)$. We assume that all ELs share the same request popularity, thus the conditional request probability for the l-th layer of video f is given by

$$p_f(l) = \begin{cases} g_{\mathrm{SDV}}(f), & \text{if } l = 1, \\ \frac{1}{L-1} g_{\mathrm{HDV}}(f), & \text{if } l = 2, 3, \ldots, L. \end{cases} \tag{2}$$

We denote the *video request probability matrix* as $\mathbf{P} \in \mathbb{R}^{F\times L}$ with its entry $P_{f,l} = p(f) \cdot p_f(l)$. Note that the exact video file popularity over a period may be learned from history data with advanced machine learning techniques at regular time intervals [10], [11].

C. Video Layer Placement and Transmission Cooperation

During off-peak hours, each SBS proactively caches video files based on the *video request probability matrix* $\mathbf{P}$. With the cached content, SBSs can thus serve most of the requests locally without using the capacity-limited backhaul during peak hours. Due to the store-and-forward design, transmission delay can be significantly reduced. Moreover, once the video is cached locally in the SBS, future repeated requests for the same video file would not incur any additional download cost over backhaul links. Now Let $\mathbf{X}^b \in \{0,1\}^{F\times L}$ denote the *video cache placement matrix* for BS b, $b \in \mathcal{B}^*$. The entry $X^b_{f,l} = 1$ if the l-th layer of video file f is stored in SBS b; and 0 otherwise. Note that the MBS is assumed to have stored the entire video library $\mathcal{F}$. Hence $X^0_{f,l} = 1$, for all f,l and its capacity is $C_0 = F \cdot L$. Due to the limited storage size of each SBS, the SBS capacity is assumed to be C_b ($C_b \ll C_0$, for all $b \in \mathcal{B}$).

Let $\mathbf{Y}^{a\to b} \in \{0,1\}^{F\times L}$ be the *video transmission cooperation matrix*. The entry $Y^{a\to b}_{f,l} = 1$ if base station b requests layer l of video file f from base station a; and 0 otherwise.

D. Cooperative Caching vs. Non-Cooperative Caching

As illustrated in Fig. 1, there are three videos stored at the server. UE 1 and UE 2 request video 1 and video 2, respectively. Each video is assumed to have three layers, including one base layer B1 and two enhancement layers, E1 and E2. Each SBS has a caching capacity up to 2. A cooperative caching mechanism is demonstrated in the transmission process of video file 1. It can be seen that UE 1 requests video

file 1 from SBS 2. Since SBS 2 only has layer E2 stored in the local cache, it requests layer B and layer E1 of video file 1 from its neighboring SBS 1. By cooperative caching, UE 1 does not need to request the missing video file layers from the MBS, which may incur a huge backhaul transmission delay. As a comparison, a noncooperative caching scheme is shown in the transmission process for video 2, which is requested by UE 2. Since SBS 3 has a local cache of layer B and layer E1 of video 2, these files can be delivered to UE 2 quickly. However, the missing layer E2 of video file 2 has to be fetched from the MBS, although nearby SBSs may have a copy of the missing file. Due to the noncooperative nature, the TTD is relatively larger.

III. Problem Formulation

In scalable video delivery, there exists a dependency between the layers of the same video file: the successful decoding of the l-th layer depends on the correct decoding of all the previous $l-1$ layers. When UE requests the lth layer, all the previous $l-1$ layers have to be requested and transmitted, and the transmission delay depends the largest delay among all the l layers. Therefore, when the BS b requests video file f with quality level l, the transmission delay $\tau_{f,l}^{b}$ can be written as

$$\tau_{f,l}^{b} = \max_{l'=1,2,\ldots,l} \left[\sum_{a} Y_{f,l'}^{a\to b} \cdot D_{a,b} \right]. \tag{3}$$

The total transmission delay (TTD) T for all the BSs is

$$T = \sum_{b=0}^{B} \sum_{f=1}^{F} \sum_{l=1}^{L} P_{f,l} \cdot \tau_{f,l}^{b}. \tag{4}$$

For a given video file popularity distribution, we aim to find an optimal caching placement schedule and BS cooperation scheme for scalable video transmissions, so that the TTD is minimized for a given SBS cache capacity and network topology. The problem is formulated as follows.

$$(\mathrm{P1}) \quad \min_{\mathbf{X}^b, \mathbf{Y}^{a\to b}} \quad T \tag{5}$$

$$\text{s.t.} \quad \sum_{f=1}^{F} \sum_{l=1}^{L} X_{f,l}^{b} \le C_b, \quad \forall b \tag{6}$$

$$X_{f,l}^{b},\ Y_{f,l}^{a\to b} \in \{0,1\}, \quad \forall a,b,f,l \tag{7}$$

$$\sum_{a=0}^{B} Y_{f,l}^{a\to b} \ge 1, \quad \forall b,f,l \tag{8}$$

$$Y_{f,l}^{a\to b} \le X_{f,l}^{a}, \quad \forall a,b,f,l, \tag{9}$$

where constraints (6) indicate that the cache content size should be no more than the cache capacity of each BS, constraints (7) ensure the variables in $\mathbf{X}^b$ and $\mathbf{Y}^{a\to b}$ take binary values, constraints (8) make sure that the video file requests initiated from one BS should be routed to one content source, and constraints (9) allow users to retrieve a content from a BS only if the content is cached there.

Problem P1 aims to identify the optimal caching placement matrix $\mathbf{X}^b$ for each BS b, and the optimal caching cooperation matrix $\mathbf{Y}^{a\to b}$ between different SBSs based on a known video file request distribution $\mathbf{P}$ and network delay topology $\mathbf{D}$. The constraints are all linear but the objective function is nonlinear (due to the maximization operation). Hence Problem P1 belongs to the nonlinear integer programming (NIP) family.

Proposition 1: Problem P1 is NP-hard.

Proof: We consider a special case where each video only contains one BL. Then the objective function (4) is a linear function w.r.t. the variables that take value 0 or 1. The constraint functions are also linear. This problem is identical to a *multi-dimensional knapsack problem*, which maximizes the sum of the values of all the items in a knapsack, so that the sum of the weights is less than or equal to the knapsack's capacity. The decision problem of the knapsack problem is NP-hard. Thus Problem P1 is NP-hard. ■

IV. Proposed Solution Approach

Since problem (5) is NP-hard, it would not be feasible to find the optimal solution within polynomial time. Hence we propose a heuristic solution to obtain a sub-optimal solution. Specifically, we relax the binary variables in (6) to continuous variables, which yields the following problem:

$$(\mathrm{P2}) \quad \min_{\mathbf{X}^b, \mathbf{Y}^{a\to b}} \quad T \tag{10}$$

$$\text{s.t.} \quad (6),\ (8),\ (9)$$

$$X_{f,l}^{b},\ Y_{f,l}^{a\to b} \in [0,1], \quad \forall a,b,f,l. \tag{11}$$

It can be seen that with such a relaxation, the objective value of problem P2 is a *lower bound* for problem P1. Although the objective function for P2 is nonlinear, it is in the form of point-wise maximum of some linear functions, and hence the objective function is convex [12, p. 68].

In this case, problem P2 is a convex problem, which can be solved by existing optimization methods (e.g., the *interior point method*) in polynomial time. In our simulations, we use the popular `gurobi` solver [13] to solve problem P2. However, since the solution to problem P2 may violate constraints (7), we propose a heuristic algorithm to obtain a feasible solution. The algorithm is presented in Algorithm 1.

The main idea of the proposed algorithm is as follows. Suppose $\widehat{\mathbf{X}}^b$ and $\widehat{\mathbf{Y}}^{a\to b}$ is the solution obtained by solving the standard convex optimization problem P2. We sort the values $\widehat{Y}_{f,l}^{a\to b}$ in a descending order. Intuitively, if the value of $\widehat{Y}_{f,l}^{a\to b}$ is large (closer to 1), it means $\widehat{Y}_{f,l}^{a\to b}$ contributes more to the objective function. Hence, we set $Y_{f,l}^{a\to b}$ to 1 as long as the capacity constraint (6) is satisfied. If file (*f,l*) is not stored in BS a at this moment, we will also set $X_{f,l}^{a}$ to 1, and the corresponding BS storage will be decreased by 1 to meet constraints (6). If $\widehat{Y}_{f,l}^{a\to b}$ is set to 1, then all the other $\widehat{Y}_{f,l}^{k\to b}$ ($k \neq a$) will be removed from set $\mathcal{T}$, so that constraints (8) are satisfied. This process will continue until all the variables are processed.

Note that the solution of Algorithm 1 is a feasible solution, since it guarantees that the capacity constraints (6) and the nonempty retrieve constraints (9) are both satisfied. Also the procedure of removal makes sure that constraints (8) are satisfied. Finally, all the element of $\widehat{\mathbf{X}}^b$ and $\widehat{\mathbf{Y}}^{a\to b}$ are binary so that constraints (7) are satisfied. The complexity of the

Algorithm 1 Proposed Algorithm

Initialize $\mathbf{X}^b \leftarrow \mathbf{0}^{F\times L}$, $\mathbf{Y}^{a\to b} \leftarrow \mathbf{0}^{F\times L}$, $\forall a, b$, and initialize $c_b \leftarrow C_b$;
Solve Problem (10) with the standard convex optimization method and obtain solution $\widehat{\mathbf{Y}}^{a\to b}$;
$\widehat{\mathbf{Y}}^{a\to b}$ forms a set $\mathcal{T}$;
while $\mathcal{T}$ is not empty **do**
    Find the maximum $\widehat{Y}^{a\to b}_{f,l}$ in set $\mathcal{T}$;
    if $c_a \geq 1$ or $X^a_{f,l} = 1$ **then**
        $Y^{a\to b}_{f,l} \leftarrow 1$;
        if $X^a_{f,l} = 0$ **then**
            $X^a_{f,l} \leftarrow 1$;
            $c_a \leftarrow c_a - 1$;
        end if
        Remove $\widehat{Y}^{k\to b}_{f,l}$, $\forall k \in \mathcal{B}^*$ from set $\mathcal{T}$;
    else
        Remove $\widehat{Y}^{a\to b}_{f,l}$ from set $\mathcal{T}$;
    end if
end while

proposed algorithm after executing the convex optimization solver is $\mathcal{O}(FLB)$.

Theorem 1: The gap between the objective function values of problems P1 and P2 tends to be zero as $FLB \to \infty$ if problem P1 is feasible.

Proof: Due to space limitation, we only provide a sketch of the proof here. The main idea is, for a real-valued function, when the number of variables is great w.r.t. the number of constraints, the duality gap from its convex relaxation (plus rounding) tends to vanish. Specifically, Let f^* and q^* denote the primal and dual optimal objective values of problem P1, respectively. Meanwhile, problem P2 is obtained by relaxing the binary constraints (7) into intervals (11). Based on Lagrangian duality, the dual problems of P1 and P2 are identical. Moreover, the optimal objective value of P2 will be q^* due to the strong convexity of P2. For P1, we claim that the duality gap will be bounded by

$$0 \leq f^* - q^* \leq \mathcal{O}\left(\frac{1}{FBL}\right). \tag{12}$$

Therefore, when $FBL \to \infty$, the difference between the objective values of P1 and P2 converges to 0. In the proof of the claim, we apply the Shapley-Folkman Thoerem [14], which states that the sum of a large number of nonconvex sets is almost convex. We refer interested readers to [15] and [16, Th. 2] for a more detailed and rigorous proof. ■

V. Simulation Results and Discussions

Consider a cellular network covering a disk area, with one MBS located at the center and nine SBSs placed on a regular grid throughout the disk area. We assume that the entire video file library is $F = 500$ and each video has $L = 3$ layers. The file size of each video layer is 1, the cache capacity of each SBS is $C_b = 20$, and the cache capacity for the MBS is sufficiently large, e.g., FL. The delay between an SBS and the MBS is set to 10. The delay between different SBSs is comparably smaller than that between an SBS and a MBS. In practice, the delay could be proportional to the distance between adjacent SBSs. In the simulations, we set $D_{a,b} = 0.2 \times |a - b|$, $\forall a, b \in \mathcal{B}$.

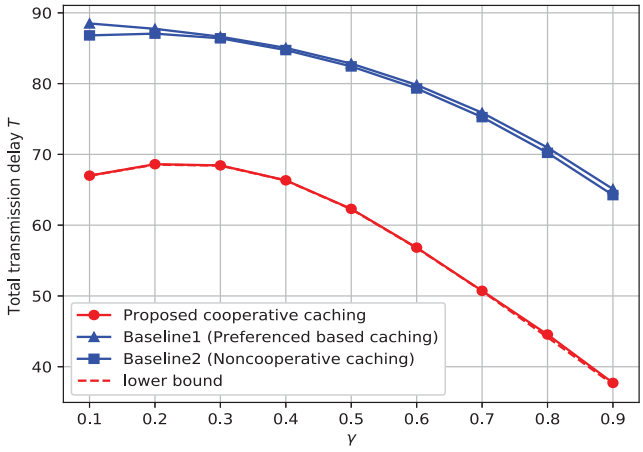

Fig. 2. The impact of the skewing parameter γ.

We consider two heuristic caching schemes as benchmarks:

- *Baseline1 (Preference based caching):* The top C_b/L most popular video files (including all the layers in that video) are cached at each SBS.
- *Baseline2 (Optimal non-cooperative caching):* Different SBSs can work in a non-cooperative way with the optimal caching placement.

Actually in Baseline1, there is no difference whether the SBSs cooperate or not, since each SBS stores the same popular video files. Its theoretical TTD can be computed as follows.

$$T_{\text{Baseline1}} = 10 \cdot (B-1) \cdot \sum_{(f,l)\in\mathcal{A}} P_{f,l}, \tag{13}$$

where $\mathcal{A}$ is a subset of the video file library that is not stored in the SBSs and has to be fetched from the MBS, which induces an additional delay of 10. Since the MBS has an entire copy of all the videos, the TD for the MBS is 0 and we multiply $(B-1)$. In Baseline2, different SBSs do not communicate with each other; to reduce the TTD, each SBS actually stores the most popular video file layers.

First of all, we investigate the performance of the proposed method under the impact of the video file popularity distribution. We assume the skewness parameter γ varies from 0.1 to 0.9. The performance of the proposed method is presented in Fig. 2. The unit of the y-axis depends on the practical measurement of the delay matrix $\mathbf{D}$. We find that Baseline1 achieves a similar performance as Baseline2. This is because under this simulation setting, the video layer popularity is quite close to the entire video file popularity. However, caching popular video layers is more intricate, hence the performance of Baseline2 is slightly better than Baseline1. Second, compared with the two baseline methods, the proposed cooperative caching scheme significantly reduces the TTD as with the increase of γ. Note that a larger γ means that most people request some popular videos. This indicates that preference based caching is more beneficial when there is a higher degree of requests. Finally, the lower bound is obtained by solving the

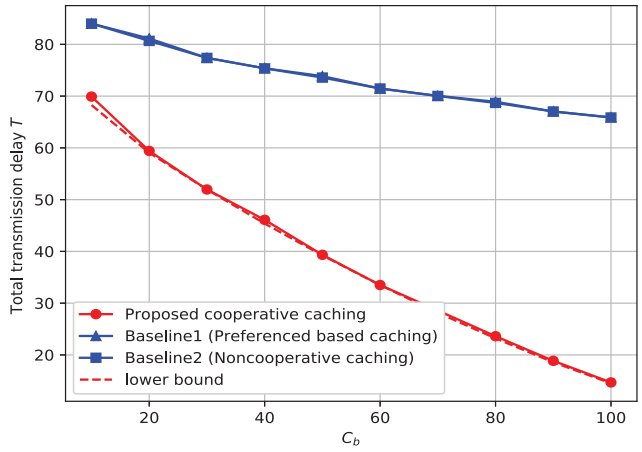

Fig. 3. The effect of the caching capacity C_b at each SBS.

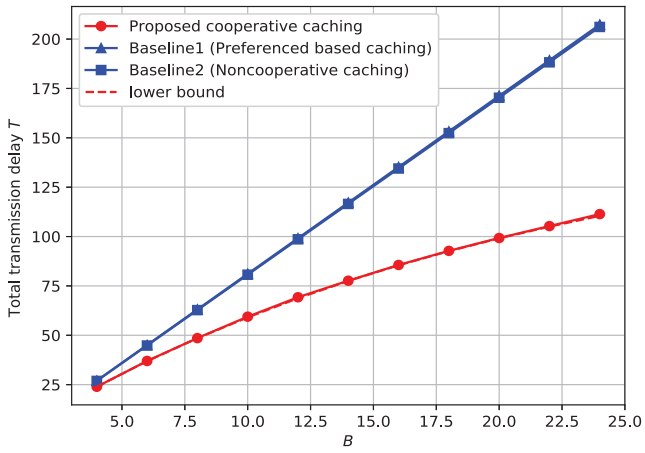

Fig. 4. The effect of the number of BSs B.

convex problem P2, although it may not be feasible due to the binary integer constraints (7). We see that with the proposed scheme the gap between the optimal lower bound and our proposed algorithm is almost zero.

Next, we fix the skew parameter γ to 0.56 [17] and the number of BSs is fixed to 10. We vary the cache capacity of each SBS. It can be seen from Fig. 3 that with the increase of each SBS, the TTD of the proposed method is reduced significantly. The performance gap between the proposed method and the baseline methods becomes larger as C_b is increased. This experiment demonstrates the superiority of the proposed method, especially when the caching capacity of SBS is large. Again, the proposed scheme achieves a very close performance to the optimal lower bound.

Finally, we fix the skew parameters to be $\gamma = 0.56$, the SBS caching capacity $C_b = 20$, and vary the number of BSs B in this region. The simulation results are presented in Fig. 4. When the number of BSs is small, the performance gaps between the three algorithms are small. This indicates that cooperative caching does not make a big difference on the network-wide performance. However, as the number of BSs is increased, the TTD of the two baseline algorithms grow linearly, while the TTD of the proposed algorithm grow quite slowly. The gain becomes larger as B is increased. The proposed method significantly reduces the TTD than the baselines and its performance is very close to the lower bound.

VI. Conclusion

In this letter, we addressed the problem of cooperative mobile edge caching for scalable video streaming in HetNets. We proved that the formulated problem is NP-hard, and then proposed a heuristic solution, which was proved to be asymptotically optimal. Simulation result demonstrated that the proposed method significantly outperformed two benchmark schemes. The TTD performance could be enhanced by a larger skewness parameter, a larger local SBS cache size, and deployment of more SBSs.

References

[1] Cisco. *The Zettabyte Era: Trends and Analysis*. Accessed: Apr. 26, 2019. [Online]. Available: https://www.cisco.com/c/en/us/solutions/collateral/service-provider/visual-networking-index-vni/vni-hyperconnectivity-wp.html

[2] K. Shanmugam, N. Golrezaei, A. G. Dimakis, A. F. Molisch, and G. Caire, "FemtoCaching: Wireless content delivery through distributed caching helpers," *IEEE Trans. Inf. Theory*, vol. 59, no. 12, pp. 8402–8413, Dec. 2013.

[3] "Advanced video coding for generic audio-visual services," Int. Telecommun. Union, Geneva, Switzerland, ITU Recommendation H.264, 2003.

[4] S. Hasan, S. Gorinsky, C. Dovrolis, and R. K. Sitaraman, "Trade-offs in optimizing the cache deployments of CDNs," in *Proc. IEEE INFOCOM*, Toronto, ON, Canada, Apr./May 2014, pp. 1–9.

[5] C. Zhan and Z. Wen, "Content cache placement for scalable video in heterogeneous wireless network," *IEEE Commun. Lett.*, vol. 21, no. 12, pp. 2714–2717, Dec. 2017.

[6] X. Zhang, T. Lv, and S. Yang, "Near-optimal layer placement for scalable videos in cache-enabled small-cell networks," *IEEE Trans. Veh. Technol.*, vol. 67, no. 9, pp. 9047–9051, Sep. 2018.

[7] J. Jiang *et al.*, "VIA: Improving Internet telephony call quality using predictive relay selection," in *Proc. ACM SIGCOMM*, Florianópolis, Brazil, Aug. 2016, pp. 286–299.

[8] L. Breslau, P. Cao, L. Fan, G. Phillips, and S. Shenker, "Web caching and Zipf-like distributions: Evidence and implications," in *Proc. IEEE INFOCOM*, New York, NY, USA, Mar. 1999, pp. 126–134.

[9] L. Wu and W. Zhang, "Caching-based scalable video transmission over cellular networks," *IEEE Commun. Lett.*, vol. 20, no. 6, pp. 1156–1159, Jun. 2016.

[10] Z. Chang, L. Lei, Z. Zhou, S. Mao, and T. Ristaniemi, "Learn to cache: Machine learning for network edge caching in the big data era," *IEEE Wireless Commun.*, vol. 25, no. 3, pp. 28–35, Jun. 2018.

[11] H. Pang, J. Liu, X. Fan, and L. Sun, "Toward smart and cooperative edge caching for 5G networks: A deep learning based approach," in *Proc. IEEE/ACM IWQoS*, Banff, AB, Canada, Jun. 2018, pp. 1–6.

[12] S. Boyd and L. Vandenberghe, *Convex Optimization*. Cambridge, U.K.: Cambridge Univ. Press, 2004.

[13] *GUROBI Optimization*. Accessed: Apr. 26, 2019. [Online]. Available: http://www.gurobi.com/index

[14] I. Ekeland and R. Temam, *Convex Analysis and Variational Problems*. Philadelphia, PA, USA: SIAM, 1999.

[15] J.-P. Aubin and I. Ekeland, "Estimates of the duality gap in non-convex optimization," *Math. Oper. Res.*, vol. 1, no. 3, pp. 225–245, Aug. 1976.

[16] L. Xiang, D. W. K. Ng, R. Schober, and V. W. S. Wong, "Cache-enabled physical layer security for video streaming in backhaul-limited cellular networks," *IEEE Trans. Wireless Commun.*, vol. 17, no. 2, pp. 736–751, Feb. 2018.

[17] M. Zink, K. Suh, Y. Gu, and J. Kurose, "Characteristics of YouTube network traffic at a campus network—Measurements, models, and implications," *Comput. Netw.*, vol. 53, no. 4, pp. 501–514, Mar. 2009.