Distributed Cooperative Caching for Utility Maximization of VoD Systems

Konstantin Avrachenkov, Jasper Goseling, Berksan Serbetci

Distributed Cooperative Caching for Utility Maximization of VoD Systems

Konstantin Avrachenkov
INRIA
Sophia Antipolis, France
konstantin.avrachenkov@inria.fr

Jasper Goseling
Mathematics of Operations Research
University of Twente, The Netherlands
j.goseling@utwente.nl

Berksan Serbetci*
EURECOM
Sophia Antipolis, France
serbetci@eurecom.fr
*Corresponding author.

***Abstract*—We consider caching of VoD contents in a cellular network in which each base station is equipped with a cache. Videos are partitioned into chunks according to a layered coding mechanism and the goal is to place chunks in caches such that the expected utility is maximized. The utility depends on the quality at which a user is requesting a file and the chunks that are available. We impose alpha-fairness across files and qualities. We develop a distributed asynchronous algorithm for deciding which chunks to store in which cache.**

I. Introduction

A promising means to increase the efficiency of Video-on-Demand (VoD) services in a cellular network is to proactively *cache* data in the base stations: part of the popular content will be stored at the base stations and the backhaul will be used only when the stored content is refreshed preferably at off-peak hours. The challenge is then to optimally place content in the base stations.

In this paper, we look at the problem from a utility perspective and develop a low-complexity distributed and asynchronous content placement algorithm for VoD systems. More specifically, we consider a layered coding (LC) mechanism, enabling to serve videos at different qualities. Users request content at a specified quality and the resulting utility is inversely proportional to the number of bits that need to be downloaded over the backhaul due to non-cached layers for the requested quality. The rationale is that latency, and therefore, quality of experience, is proportional to the size of this download. We impose alpha-fairness across files and qualities [12]. We provide an algorithm that optimizes the content placement in caches while keeping the communication between caches in the network to a minimum as the caches need to exchange information only with their neighbours, *i.e.*, with those whom they have overlapping coverage area with.

Next, we provide a brief discussion work that is most closely related to the current paper. In [6], Dehghan *et al.* have provided a utility-driven caching for a single cache, where each content has been associated with a utility, which is a function of the corresponding content hit probability. In [3] we have developed a distributed asynchronous algorithm that deals with the miss probability minimization by casting the problem into the framework of potential games for general networks, and showed that our algorithm converges to a Nash equilibrium, and in fact to the best Nash equilibrium in most practical scenarios. In [4] the developed algorithm has been used for the residual bandwidth minimization of VoD systems with several video partitioning mechanisms. In [1], Applegate *et al.* have presented an approach for optimal content placement for a large-scale VoD by formulating the problem as a mixed integer problem. The main difference between the present work; and [6] is the consideration of a network with many caches instead of a single cache, and [3], [4] is the considered performance measure, and [1] is the concept of video partitioning and the heterogeneity of the video popularities over the network. See [3], [10] for a more exhaustive review of the existing caching work.

Our main contributions in this paper are as follows:

- We use layered coding (LC) mechanism to divide VoD contents into chunks and consider alpha-fair utility functions to represent a versatile notion of fairness for user satisfaction throughout the whole network;
- We provide a distributed asynchronous algorithm which can be interpreted as giving the best response dynamics in a potential game.
- We provide the optimal solution to the best response dynamics via Lagrangian;
- We evaluate our algorithm through numerical examples. We study the optimal content placement probabilities for different degrees of fairness.

Let us outline the organization of the paper. In Section II we give the formal model and problem definitions. In Section III we provide the game formulation of the problem, analyze the structures of the best response dynamics and Nash equilibria. In Section IV, we present practical implementations of our low-complexity algorithm and show the resulting optimal content placements for different notions of fairness of utility in a real network.

II. Model and Problem Definition

We consider a network of N base stations that are located in the plane $\mathbb{R}^2$. We will use the notation $[1:N]=\{1,\dots,N\}$ and $\Theta = \mathbb{P}([1:N]) \setminus \emptyset$, where $\mathbb{P}([1:N])$ is the power set of $[1:N]$. We specify the geometric configuration of the network through A_s, $s \in \Theta$, which denotes the area of the plane that is covered only by the caches in subset s, namely $A_s = (\cap_{\ell \in s} \bar{A}_\ell) \cap (\cap_{\ell \notin s} \bar{A}_\ell^c)$, where $\bar{A}_\ell$ is the complete coverage region of cache ℓ.

As a special case we will consider the case that all base stations have the same circular coverage region with radius r. In this case we specify the location of each base station, with x_m for the location of base station $m \in [1:N]$. We then obtain $\bar{A}_m$ as the disc of radius r around x_m.

Each base station is equipped with a cache that can be used to store videos from a content library $\mathcal{C}_v = \{c_1, \ldots, c_J\}$, where $J < \infty$. Each element c_j represents a video, where j is the file (video) index. The size of video c_j is denoted by w_j.

Next, videos are partitioned into chunks. We assume that video c_j is partitioned into Q chunks and consequently, the chunk library consists of video chunks $\mathcal{C}_c = \{c_{1,1}, \ldots, c_{1,Q}, \ldots, c_{J,1}, \ldots, c_{J,Q}\}$, where $J, Q < \infty$. Each element $c_{j,q}$ represents a video chunk, where j is the file (video) index and q is the chunk index. The chunk size for the element $c_{j,q}$ is denoted by $w_{j,q}$.

The motivation behind partitioning videos into chunks is as follows. In this work we will consider LC mechanism [14]. If the video is partitioned by using LC mechanisms (e.g., MPEG-2, MPEG-4, etc.), the video chunks represent either the base layer or one of the enhancement layers. The base layer is necessary for the media stream to be decoded. Accordingly, the further enhancement layers are applied to improve the video quality. Hence, for the layered coding mechanism, the element $c_{j,1}$ represents the base layer for video j, and $c_{j,2}$ is the first enhancement layer required to obtain a better video quality, and so forth. For the LC mechanism, the chunk size $w_{j,q}$ will then depend on the average base size of the different video qualities, and the total size of the video j with the highest video quality is equal to

$$w_j = \sum_{q=1}^{Q} w_{j,q}. \tag{1}$$

Our interest is in users located in the plane over the area that is covered by the base stations, *i.e.*, uniformly distributed in $A_{cov} = \cup_{s\in\Theta} A_s$. The probability of a user in the plane being covered by caches $s \in \Theta$ (and is not covered by additional caches) is denoted by $p_s = |A_s|/|A_{cov}|$. A user located in A_s, $s \in \Theta$ can connect to all caches in subset s, and has access to all video chunks stored in these caches.

We assume that there are R video quality levels. The probability that video j is requested with video quality ρ by a user at $x_u \in s$ is denoted by $a_{j,\rho,s}$. Here $\rho = 1$ refers to the video with the lowest quality and $\rho = R$ refers to the case where the video has the highest quality. There is an obvious relation between the required chunks and the desired video quality. For the LC mechanism, files can be divided into $Q = R$ layers with the proper size selections ($w_{j,q}$) and receiving all layers gives the video with the highest quality.

We assume that the distribution for the requested video's quality depends on the user's location, and the requested video quality for a user located in $s \in \Theta$ follows a known and fixed probability mass function (pmf) $f^{(s)}(\rho)$.

For the file popularities, we assume that $a_1 \geq a_2 \geq \cdots \geq a_J$. We assume that video popularities do not vary as ρ changes. Even though any popularity distribution can be used, most of our numerical results will be based on the Zipf distribution for the video popularities for any video quality ρ. In that case, for any video quality indicator ρ, the probability that a user will ask for content j is equal to

$$a_j = \frac{j^{-\gamma}}{\sum_{j=1}^{J} j^{-\gamma}}, \tag{2}$$

where $\gamma > 0$ is the Zipf parameter.

Since a_j and $f^{(s)}(\rho)$ represent statistically independent random variables, we conclude that the probability that video j is requested by a user located at $x_u \in s$ with video quality ρ is equal to

$$a_{j,\rho,s} = \frac{j^{-\gamma}}{\sum_{j=1}^{J} j^{-\gamma}} f^{(s)}(\rho). \tag{3}$$

Content is placed in caches using knowledge of the request statistics $a_{j,\rho,s}$, but without knowing the actual request made by the user. We denote the placement policy for cache m as

$$b_{j,q}^{(m)} := \begin{cases} 1, & \text{if } c_{j,q} \text{ is stored in cache } m, \\ 0, & \text{if } c_{j,q} \text{ is not stored in cache } m, \end{cases} \tag{4}$$

and the overall placement strategy for cache m as

$$\mathbf{B}^{(m)} = \begin{bmatrix} b_{1,1}^{(m)} & \cdots & b_{J,1}^{(m)} \\ \vdots & \ddots & \vdots \\ b_{1,Q}^{(m)} & \cdots & b_{J,Q}^{(m)} \end{bmatrix}$$

as a $Q \times J$ matrix. The overall placement strategy for the network is denoted by $\mathbf{B} = \left[\mathbf{B}^{(1)}; \ldots; \mathbf{B}^{(N)}\right]$ as an $Q \times J \times N$ three-dimensional matrix.

Caches have capacity K, *i.e.*,

$$\sum_{j=1}^{J} \sum_{q=1}^{Q} w_{j,q} b_{j,q}^{(m)} \leq K, \forall m.$$

For clarity of presentation, we assume homogeneous capacity for the caches. However, our work can immediately be extended to the network topologies where caches have different capacities (*i.e.*, for the case where cache $m \in [1:N]$ has capacity K_m).

Next,we are interested in designing a cache placement strategy that optimizes the sum of utilities over all files throughout the network for LC mechanism.

We will use α-fair utility functions, each yielding different notions of fairness [13]. Then, the sum of utilities over all files throughout the network is given by

$$U(\mathbf{B}) = \sum_{s\in\Theta} \sum_{j=1}^{J} \sum_{\rho=1}^{Q} p_s a_{j,\rho,s} \frac{h_{j,\rho,s}^{1-\alpha_{\rho,s}}}{1-\alpha_{\rho,s}}, \tag{5}$$

where $\alpha_{\rho,s}$ is the fairness parameter for the videos requested with quality ρ by the users located in subregion $s \in \Theta$, and

$$h_{j,\rho,s} = \frac{\sum_{q=1}^{\rho} w_{j,q} \left[1 - \prod_{\ell\in s} \left(1 - b_{j,q}^{(\ell)}\right)\right]}{\sum_{q=1}^{\rho} w_{j,q}}. \tag{6}$$

Here $h_{j,\rho,s}$ defines the portion of the requested video j with video quality ρ that is available to the user located in $s \in \Theta$. Since the data available at the cache do not give rise to any load in the backhaul, the transmission rate has a direct relation with this parameter and the content stored in caches will be transmitted to the user with lower latency.

Our goal is to find the optimal placement strategy maximizing the sum of utilities of the contents throughout the overall network as follows:

Problem 1.

$$\max \; U(\mathbf{B})$$
$$\mathbf{s.t.} \quad \sum_{j=1}^{J}\sum_{q=1}^{Q} w_{j,q} b_{j,q}^{(m)} \leq K, \; \forall m, \tag{7}$$
$$b_{j,q}^{(m)} \in [0,1], \; \forall j,q,m. \tag{8}$$

It is easy to verify that Problem 1 is not concave. We will provide a distributed asynchronous algorithm to address Problem 1 in which we iteratively update the placement policy at each cache. In [3], [4] we have showed that we can define an algorithm that can be viewed as the best response dynamics in a potential game for similar non-concave problems. We will present a similar algorithm here. We make use of the following notation. Denote by $\mathbf{B}^{(-m)}$ the placement policies of all caches except cache m. We will write $U(\mathbf{B}^{(m)}, \mathbf{B}^{(-m)})$ to denote $U(\mathbf{B})$. Also, for the sake of simplicity for the potential game formulation that will be presented in the following section, let $U^{(m)}$ denote the sum of utilities of the users within coverage region of cache m, *i.e.*,

$$U^{(m)}\left(\mathbf{B}^{(m)}, \mathbf{B}^{(-m)}\right) = \sum_{\substack{s\in\Theta \\ m\in s}} \sum_{j=1}^{J}\sum_{\rho=1}^{R} p_s a_{j,\rho,s} \frac{h_{j,\rho,s}^{1-\alpha_{\rho,s}}}{1-\alpha_{\rho,s}}$$
$$= \sum_{\substack{s\in\Theta \\ m\in s}} \sum_{j=1}^{J}\sum_{\rho=1}^{R} p_s a_{j,\rho,s} \frac{\left(\frac{\sum_{q=1}^{\rho} w_{j,q}\left[1-\left(1-b_{j,q}^{(m)}\right)\zeta^{(m)}(j,q)\right]}{\sum_{q=1}^{\rho} w_{j,q}}\right)^{1-\alpha_{\rho,s}}}{1-\alpha_{\rho,s}},$$
$$= \sum_{j=1}^{J}\sum_{q=1}^{Q}\sum_{\substack{s\in\Theta \\ m\in s}} p_s a_{j,q,s} \frac{\left(\frac{\sum_{t=1}^{q} w_{j,t}\left[1-\left(1-b_{j,t}^{(m)}\right)\zeta^{(m)}(j,t)\right]}{\sum_{t=1}^{q} w_{j,t}}\right)^{1-\alpha_{\rho,s}}}{1-\alpha_{\rho,s}}$$
$$= \sum_{j=1}^{J}\sum_{q=1}^{Q}\sum_{\substack{s\in\Theta \\ m\in s}} p_s a_{j,q,s} \frac{\eta_{j,q,s}^{1-\alpha_{\rho,s}}}{1-\alpha_{\rho,s}} \tag{9}$$

where

$$\eta_{j,q,s} = \frac{\sum_{t=1}^{q} w_{j,t}\left[1-\left(1-b_{j,t}^{(m)}\right)\zeta^{(m)}(j,t)\right]}{\sum_{t=1}^{q} w_{j,t}}, \tag{10}$$

and

$$\zeta^{(m)}(j,q) = \prod_{\ell\in s\setminus\{m\}} \left(1-b_{j,q}^{(\ell)}\right). \tag{11}$$

III. Potential Game Formulation

In this section we provide a distributed asynchronous algorithm to address Problem 1 in which we iteratively update the placement policy at each cache. We will show that this algorithm can be formulated as providing the best response dynamics in a potential game.

In our algorithm, each cache tries selfishly to maximize the utilities of the users within its coverage region $U^{(m)}$ defined in (9). Given a placement $\mathbf{B}^{(-m)}$ by the other caches, cache m solves for $\mathbf{B}^{(m)}$ in

Problem 2.

$$\max \; U^{(m)}\left(\mathbf{B}^{(m)}, \mathbf{B}^{(-m)}\right)$$
$$\mathbf{s.t.} \quad \sum_{j=1}^{J}\sum_{q=1}^{Q} w_{j,q} b_{j,q}^{(m)} \leq K, \tag{12}$$
$$b_{j,q}^{(m)} \in [0,1], \; \forall j,q. \tag{13}$$

Each cache continues to optimize its placement strategy until no further improvements can be made. At this point $\mathbf{B}$ is a *Nash equilibrium strategy* that maximizes the overall utility satisfying

$$U^{(m)}(\mathbf{B}^{(m)}, \mathbf{B}^{(-m)}) \geq U^{(m)}(\bar{\mathbf{B}}^{(m)}, \bar{\mathbf{B}}^{(-m)}), \; \forall m, \mathbf{B}^{(m)}. \tag{14}$$

We will refer to this game as the *content placement game* and demonstrate in the next subsections that this game is a potential game [9] with many nice properties.

A. Convergence analysis

In this section, we will prove for that if the caches repeatedly update their placement strategies it is guaranteed to converge to a Nash equilibrium in finite time. Note that the order of the updates is not important as long as all caches are scheduled infinitely often.

Theorem 1. *The content placement game defined by payoff function* (9) *is a potential game with the potential function given in* (5)*. If we schedule each cache infinitely often, the best response dynamics converges to a Nash equilibrium in finite time.*

Proof. In order to show that the game is a potential with the potential function $U(\mathbf{B})$, it is easy to verify that

$$U^{(m)}\left(\bar{\mathbf{B}}^{(m)}, \mathbf{B}^{(-m)}\right) - U^{(m)}\left(\mathbf{B}^{(m)}, \mathbf{B}^{(-m)}\right)$$
$$= U\left(\bar{\mathbf{B}}^{(m)}, \mathbf{B}^{(-m)}\right) - U\left(\mathbf{B}^{(m)}, \mathbf{B}^{(-m)}\right),$$

which completes the proof of the first statement. The proof states that the improvement in the utility by the best response after each update is equal to the improvement in the utility in the overall network. The detailed analysis is trivial and skipped due to space constraints.

Now, since there exists only a finite number of placement strategies, none of the caches will be missed in the long-run. Moreover, each non-trivial best response provides a positive

improvement in the potential function in a potential game. Hence, we are guaranteed to converge to a Nash equilibrium in finite time for both games. □

B. Structure of the best response dynamics

In this subsection we will analyze the structure of the best response dynamics. We will show that solution to Problem 2 can be obtained by solving a concave optimization problem and we provide the general solution for different fairness parameters $\alpha_{\rho,s}$ via Lagrangian duality.

Since the feasible solution set to Problem 2 is convex and the objective function is strictly concave and continuous, the optimal solution exists and this optimal solution is a unique maximizer of the function. We can write the Lagrangian function as

$$L(\mathbf{B},\lambda,\mathbf{M},\mathbf{N}) = U^{(m)}\left(\mathbf{B}^{(m)},\mathbf{B}^{(-m)}\right) - \lambda\left(\sum_{j=1}^{J}\sum_{q=1}^{Q} w_{j,q}b_{j,q}^{(m)} - K\right) + \sum_{j=1}^{J}\sum_{q=1}^{Q}\mu_{j,q}b_{j,q}^{(m)} - \sum_{j=1}^{J}\sum_{q=1}^{Q}\nu_{j,q}\left(b_{j,q}^{(m)}-1\right),$$

where $\mathbf{B}^{(m)},\mathbf{M},\mathbf{N}\in\mathbb{R}_{+}^{J\times Q}$, and $\lambda\in\mathbb{R}$.

In order to achieve the maximum in $L(\mathbf{B},\lambda,\mathbf{M},\mathbf{N})$, $\mathbf{B}^{(m)}$ must satisfy

$$\frac{\partial U^{(m)}\left(\mathbf{B}^{(m)},\mathbf{B}^{(-m)}\right)}{\partial b_{j,q}^{(m)}} - \lambda w_{j,q} + \mu_{j,q} - \nu_{j,q} = 0, \quad (15)$$

which is easy to solve since the Lagrangian is separable with respect to $b_{j,q}^{(m)}$.

From the nature of the problem, $b_{j,q}^{(m)}$'s can take values between 0 and 1; which yields to a probabilistic placement in caches. This yields to the conclusion that one way of possibly storing chunks in the caches is by using the obtained optimal probabilities based on the optimal solution at the off-peak hours [2], [5], [11]. Another strategy is to use time-to-live (TTL) caches [7], [6] by setting timers to control the storage probabilities of different chunks.

C. Structure of Nash equilibria

In this subsection we provide insight into the structure of the Nash equilibria of the content placement game. We know from the previous subsection that the games is a potential game. Hence, the Nash equilibria for the game corresponds to the optimal placement strategies satisfy the solutions of the dual problem of Problem 1.

Corollary 1. *Let $\bar{\mathbf{B}}$ denote a placement strategy at a Nash equilibrium of the content placement game. Then, $\bar{\mathbf{B}}$ satisfies the solution obtained via* (15) *for any $\alpha_{\rho,s}$ and $\forall m = 1,\ldots,N$.*

Algorithm 1: Random Order Best Response (ROBR)

```
initialize B^(m) = 0_{Q,J}, ∀m ∈ [1 : N];
set imp(m) = 1, ∀m ∈ [1 : N] ;
set imp = [imp(1), ..., imp(N)];
while imp ≠ 0 do
    m = Uniform(N);
    Set imp(m) = 0;
    Solve Problem 2 for cache m and find B̄^(m) using
     the information coming from neighbours;
    Compute U^(m)(B̄^(m), B^(-m));
    if U^(m)(B̄^(m), B^(-m)) − U^(m)(B^(m), B^(-m)) ≠ 0
     then
        imp(m) = 1
    end
end
```

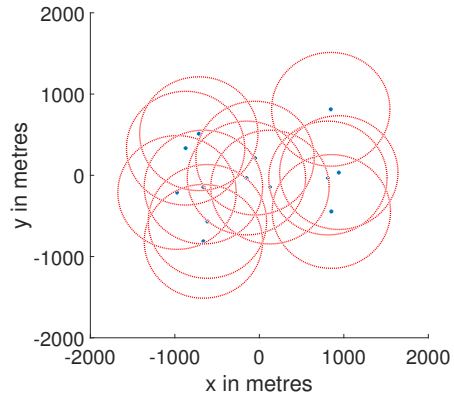

Fig. 1. Locations of Base Stations from OpenMobileNetwork dataset.

D. A real wireless network: Berlin network

In this section we will evaluate our theoretical results for the topology of a real wireless network. We have taken the positions of the base stations provided by the OpenMobileNetwork project [15]. The base stations are located in the area 974×812 ms around the TU-Berlin campus. We will consider these base stations as our caches with certain cache capacities. The coverage radius of the base stations is equal to $r = 700$ m. The locations of the base stations and their corresponding coverage areas are shown in Figure 1.

IV. Performance Evaluation

In this section we will present practical implementations of our algorithm for the content placement game and evaluate our theoretical results according to a network of caches with their geographical locations following a real wireless network.

A. The ROBR algorithm

We will use the Random Order Best Response (ROBR) algorithm [3] for the content placement game to maximize the user utilities. The basic idea of our algorithm is to repeatedly perform best response dynamics presented in Section III-B.

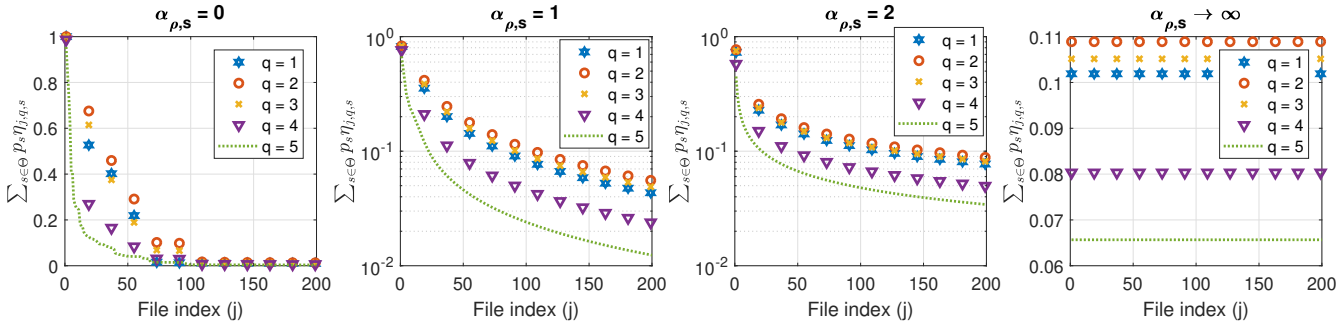

Fig. 2. The cached portions of the chunks for different notions of fairness.

TABLE I
VIDEO QUALITY SPECIFICATIONS

Video quality (ρ)	Base Size (MB/min)	Video Size (GB)
1 (240p)	2.56	0.1024
2 (360p)	4.30	0.1720
3 (480p)	6.79	0.2716
4 (720p)	15.49	0.6196
5 (1080p)	32.70	1.3080

For ROBR algorithm, at each iteration step, a random cache is chosen uniformly from the set $[1 : N]$ and updated by applying best response dynamics. We assume that all caches are initially empty (*i.e.*, $\mathbf{B}^{(m)} = 0_{Q,J}$, where $0_{Q,J}$ represents a $Q \times J$ zero matrix, $\forall m \in [1 : N]$.). The algorithm stops when $U^{(m)}\left(\mathbf{B}^{(m)}, \mathbf{B}^{(-m)}\right)$ converges. ROBR algorithm is shown in Algorithm 1.

We consider the content library of size $J = 200$. We assume a Zipf distribution for the video popularities, setting $\gamma = 1$ and taking a_j according to (2). We consider the case where videos have $R = 5$ different video qualities. We use the video quality bandwidth requirements and the base size data given in [8] for the chunk sizes, where the base size is the average of the sizes of a large set of tracked videos in certain video qualities given in megabytes per minute (MB/min). We assume that videos in the content library are all 40 minutes long. The corresponding video quality specifications are shown in Table I. The first column indicates the video quality index ρ, the second column indicates the base sizes of the videos and the third column indicates the sizes of the 40 minutes long videos.

The videos are partitioned into $Q = 5$ chunks by using LC mechanism. From the third column of Table I, it immediately follows that $w_{j,1} = 102.4$ MB, $w_{j,2} = 69.6$ MB, $w_{j,3} = 99.6$ MB, $w_{j,4} = 348.0$ MB, and $w_{j,5} = 688.4$ MB, $\forall j \in [1, J]$. Finally, we set $K = 6.54$ GB, *i.e.*, each cache can store five 1080p movies.

Throughout the network we use a uniform distribution for the requested video qualities, *i.e.*, $f^{(s)}(\rho) = 1/5$, $\rho \in 1, 2, 3, 4, 5$.

In Figure 2 the cached portions of the chunks throughout the whole network is shown. For $\alpha_{\rho,s} = 0$, the optimal placement strategy is designed in order to minimize the residual bandwidth. Therefore, both sizes of the chunks and the file popularities are affecting the optimal placement. As $\alpha_{\rho,s}$ increases, the effect of video popularities on the optimal solution becomes less effective; in fact, when $\alpha_{\rho,s} \to \infty$, the video popularities becomes completely ineffective and the chunks for different videos at the same layer are all stored with equal probability, yielding max-min fairness. Note that in this case the chunk sizes are still taken into account since the chunk sizes are already embedded in $h_{j,\rho,s}$.

REFERENCES

[1] D. Applegate, A. Archer, V. Gopalakrishnan, S. Lee, and K. K. Ramakrishnan, "Optimal content Placement for a large-scale VoD system," *IEEE/ACM Trans. on Networking*, vol. 24, no. 4, pp. 2114–2127, 2016.

[2] K. Avrachenkov, X. Bai, and J. Goseling, "Optimization of caching devices with geometric constraints," *International Journal of Performance Evaluation*, vol. 113, pp. 68–82, August 2017.

[3] K. Avrachenkov, J. Goseling, and B. Serbetci, "A low-complexity approach to distributed cooperative caching with geographic constraints," *Proceedings of the ACM on Measurement and Analysis of Computing Systems (POMACS)*, vol. 1, no. 1, pp. 1–25, June 2017.

[4] K. Avrachenkov, J. Goseling, and B. Serbetci, "Distributed cooperative caching for VoD with geographic constraints," in *Proceedings of IEEE WiOpt 2019*, also *arXiv: 1903.02406*, 2019.

[5] B. Błaszczyszyn, and A. Giovanidis, "Optimal geographic caching in cellular networks," *IEEE International Conference on Communications (ICC) 2015*, pp. 3358–3363, London, UK, June 2015.

[6] M. Dehghan, L. Massoulie, D. Towsley, D. Menasche, and Y. C. Tay, "A utility optimization approach to network cache design," *IEEE INFOCOM 2016*, San Francisco, USA, April 2016.

[7] N. C. Fofack, P. Nain, G. Neglia, and D. Towsley, "Performance evaluation of hierarchical TTL-based cache networks," *Computer Networks*, vol. 65, pp. 212–231, 2014.

[8] D. K. Krishnappa, D. Bhat, and M. Zink, "DASHing YouTube: An analysis of using DASH in YouTube video service," *38th Annual IEEE Conf. on Local Comp. Networks*, pp. 407–415, Sydney, Australia, 2013.

[9] D. Monderer, and L. Shapley, "Potential games," *Games and Economic Behavior*, vol. 14, pp. 124–143, 1996.

[10] G. S. Paschos, G. Iosifidis, M. Tao, D. Towsley, and G. Caire, "The role of caching in future communication systems and networks," *IEEE J. on Sel. Areas in Comm.*, vol. 36, no. 6, pp. 1111–1125, 2018.

[11] B. Serbetci, and J. Goseling, "Optimal geographical caching in heterogeneous cellular networks," *arXiv: 1710.09626*, 2017.

[12] R. Srikant, *The mathematics of internet congestion control*. Springer Pr., 2004.

[13] R. Srikant, and L. Ying, *Communication networks: An optimization, control, and stochastic networks perspective*. Cambridge Univ. Pr., 2013.

[14] T. Wiegand, G.J. Sullivan, G. Bjontegaard, and A. Luthra, "Overview of the H.264/AVC video coding standard," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 13, no. 7, pp. 560–576, 2003.

[15] A. Uzun, *Semantic Modeling and Enrichment of Mobile and WiFi Network Data*. Springer Pr., 2019.