Proactive Wireless Caching at Mobile User Devices for Energy Efficiency

Arif Can Güngör
Department of Electrical and Electronics Engineering
Middle East Technical University
Ankara, Turkey
Email: can.gungor@metu.edu.tr

Deniz Gündüz
Department of Electrical and Electronic Engineering
Imperial College London
London, UK
Email: d.gunduz@imperial.ac.uk

***Abstract*—Proactive content caching** **at user terminals is studied from an energy efficiency perspective. Assuming that the variable-rate demands of a user can be predicted accurately over a certain time period, the optimal transmission strategy that minimizes the total energy consumption is characterized. The reduction in the energy consumption is obtained both by increasing the total transmission time of a request, and by downloading it at better channel conditions, rather than downloading it at the time of use. Both gains are possible thanks to the limited cache memory at the user device, in which the pre-downloaded content is stored until it is requested by the application layer, such as a video player. We formulate the optimal proactive transmission strategy as the solution of a convex optimization problem, and evaluate the minimum total energy requirement numerically. We also provide a *backward water-filling* interpretation for the optimal caching strategy.**

I. Introduction

A significant portion of the growing mobile data traffic is caused by streaming pre-stored video content, such as YouTube, MUBI, Netflix videos [1]. Streaming applications require high-bandwidth connection to the network, and impose additional delay constraints to avoid interruptions on the user side. The growing demand is typically addressed by increasing the achievable data rates. On the other hand, the access points which users connect to are shared by more and more users, and their limited backhaul links are becoming more and more congested. Additionally, most of the streaming content is accessed through mobile devices with limited batteries, which puts additional constraints on the data rates that can be achieved even in low traffic periods.

Moving content to the network edge is a potential solution to these problems; it reduces backhaul bandwidth requirements, and improves users' perceived quality of experience (both delay and reconstruction fidelity). Content caching at wireless access points has been studied in [3], [4], [6]. In this paper, we go one step further and consider caching directly at the user devices. Considering the low cost and wide availability of memory, *proactive caching* directly at user devices can provide significant gains for the next generation wireless networks by exploiting the available distributed storage space.

Proactive caching can provide two types of gains. First, it allows the user to download the same file over a longer period of time. For most common channel coding and modulation schemes energy required to transmit a data packet is reduced by increasing the transmission time. Second, it gives the user an additional degree-of-freedom to download the content at better channel and traffic conditions. For example, if the transmitter knows in advance which movie is going to be watched by the user, and at which desired quality, the download can start prior to the playback time, rather than being streamed in real-time. This can lead to significant energy savings, especially when the playback time coincides with a peak traffic period or a weak channel condition.

On the other hand, pre-downloaded data needs to be stored in user's cache memory, which is typically limited. Hence, only a certain level of energy saving can be achieved through proactive caching. Our goal here is to study the *fundamental performance limits of proactive caching in terms of energy efficiency under a cache capacity constraint*. We assume that all the future data request rates and channel conditions of the user are known non-causally over a given period of time, i.e., *offline optimization*. The offline solution will serve as an upper bound on the potential gains of proactive caching, and also provide heuristics for the design in more uncertain scenarios.

We consider the total energy consumption until the deadline as the measure of performance, and show that the energy minimization under a given cache capacity constraint can be formulated as a convex optimization problem. Focusing on a Gaussian channel with time-varying channel quality, the optimal solution is characterized as a *backward water-filling algorithm*. The detailed characteristics of the algorithm and numerical results are also provided.

Proactive caching has been considered previously for various other goals. Client's local buffer is exploited in [12] to reduce the rate variability in the transmission of variable-bit-rate (VBR) compressed video. Downlink of multiple VBR video streams in a cellular network is studied in [9] with the goal of maximizing the total transmitted data. In [2] the authors investigate energy-efficient downlink video transmission by predicting user download rates. In [5], by controlling the buffer in an anticipatory manner, the authors minimize the delay and the number of buffered video segments, and maximize the video quality for wireless streaming. Similarly, [10] presents proactive seeding in order to reduce the peak traffic, taking into account the background load. In [11] the number of utilized subchannels is minimized over a time-varying channel

978-1-4673-6540-6/15/$31.00

by controlling user's buffer. Proactive caching is proposed in [13] to minimize the delay for mobile users, by caching files in base stations that are located on the estimated trajectory of a user. In these works, caching is utilized to minimize lateness [5], delay [13] or peak demand [10], or maximizing video quality [5]. We , on the other hand, focus on energy efficient variable-rate content delivery over a time-varying channel. In [8] we have extended the approach here to proactive caching at access points, in which case, the proactive caching gain is combined with local caching gain thanks to the downloading of the same file by multiple users.

The system model and the problem statement are presented in Section II. We describe the optimal transmission strategy in Section III, and illustrate the water-filling interpretation first for an infinite, and then for a finite cache capacity. The numerical results for the optimal proactive caching scheme, and comparisons with reactive caching are presented in Section IV, followed by conclusions in Section V.

II. System Model

We consider variable-rate content requests corresponding to different media types and qualities. We model these requests using a slotted time framework, such that the user request rate, $d(t)$, is constant within each time slot (TS). Let $0 < t_1^r < t_2^r < \cdots < t_D^r < T$ denote the time instants at which $d(t)$ changes, where T is the end of the time frame over which we want to minimize the total energy consumption. We assume that the variations in the quality of the channel from the access point to the user are slow. Similarly to the data rates, we assume that the channel quality, $h(t)$, remains constant within a TS (not necessarily the same TSs as demand variations), and changes from one TS to the next. Let $0 < t_1^c < t_2^c < \cdots < t_H^c < T$ be the time instants at which $h(t)$ changes.

We can combine the time instants at which the channel state or the user's download rate changes, into a single time series $t_0 = 0 < t_1 < t_2 < \cdots < t_{N-1} < t_N = T$. We denote the channel power gain and the request rate of the user within TS i as h_i and d_i, respectively. That is, $h(t) = h_i$ and $d(t) = d_i$ for $t \in [t_{i-1}, t_i)$. We denote the length of TS i as τ_i, i.e., $\tau_i \triangleq t_i - t_{i-1}$, for $i = 1, 2, \ldots, N$. Note that the TSs do not necessarily have the same duration.

Instantaneous transmission rate at time t from the base station to the user is a function of the channel power gain $h(t)$ and the transmission power $p(t)$, and is given as follows[1]:

$$r(t) = \log(1 + h(t)p(t)). \tag{1}$$

We will use transmission rate and download rate interchangeably to refer to $r(t)$. We also highlight that the main results and conclusions of our paper are not dependent on the particular rate function. This will only impact the nature of the resultant optimal solution (i.e., the water-filling interpretation) and the numerical results, but these can easily be extended to other concave, non-decreasing rate-power functions.

[1]For simplicity, all logarithms are in the natural basis, and the rates are considered in nats/sec.

In conventional *reactive* streaming, the user downloads the content at the rate and time of request, i.e., $r(t) = d(t)$. With our channel model, the total energy cost of reactive transmission over the period of interest is found as

$$\sum_{i=1}^{N} \tau_i \cdot \frac{e^{d_i} - 1}{h_i}. \tag{2}$$

Our goal here is to come up with a proactive caching strategy that can potentially download content to the user in advance in order to reduce the energy consumption.

We assume that the contents are transmitted in the order they are requested by the user, and the bits that are transmitted to the user which are not requested yet are stored in the user's cache memory. Data is removed from the cache at the time they are requested by the application layer. We assume that all the user's demands must be satisfied; hence, the rates assigned to the TSs have to satisfy the following constraints:

$$\int_0^t d(u)du \leq \int_0^t r(u)du, \forall t \in [0, T]. \tag{3}$$

This constraint on the download rate guarantees that the user does not experience any outages. On the other hand, the data that has been downloaded, but that has not yet been requested by the application layer, needs to be stored in the cache memory. Since the cache memory is limited, we have the following constraint:

$$\int_0^t r(u)du - \int_0^t d(u)du \leq B, \forall t \in [0, T]. \tag{4}$$

This constraint assures that no data is lost due to the overflows in the cache memory.

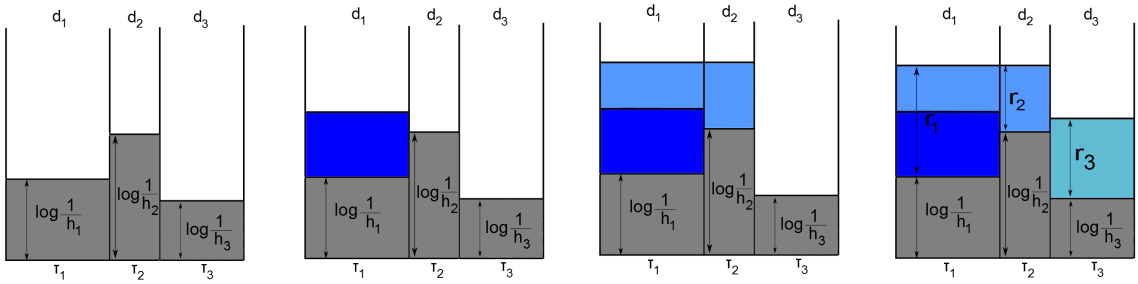
Our goal is to minimize the total energy consumption of the system over a given time frame $[0, T]$. Accordingly, the corresponding optimization problem can be written in terms of the download rates as follows:

$$\min_{r(t)\geq 0} \int_0^T \frac{e^{r(u)} - 1}{h(u)} du \tag{5}$$

$$s.t. \quad (3) \text{ and } (4). \tag{6}$$

Note that this is an infinite dimensional optimization problem, which is challenging to solve in its current form. However, it can be shown that, thanks to the slotted nature of the variations in the demand rate and the channel conditions, the dimensionality of the problem can be reduced. Since the channel gain and the user's request rate remain constant within each TS, it follows from the convexity of the objective function (5) that the optimal transmission rate and power also remain constant within a TS [14]. This means that we only need to optimize the transmission rates, r_i, or, equivalently, the transmission powers p_i, for $i = 1, 2, \ldots, N$. Hence, we can rewrite the optimization problem (5) as follows:

$$\min_{r_i \geq 0} \quad \sum_{i=1}^{N} \tau_i \frac{e^{r_i} - 1}{h_i} \tag{7}$$



(a) Channel states. (b) First request is satisfied. (c) Second request is satisfied. (d) Third request is satisfied.

Fig. 1. Illustration of the water-filling algorithm for rate allocation with an infinite cache memory.

$$\text{s.t.} \quad \sum_{i=1}^{n} \tau_i (d_i - r_i) \leq 0, \text{for} \quad n = 1, \ldots, N,$$
$$\sum_{i=1}^{n} \tau_i (r_i - d_i) - B \leq 0, \text{for} \quad n = 1, \ldots, N.$$

It is not hard to see that (7) is a convex optimization problem with a convex objective function and linear constraints. Hence, it can be numerically solved in polynomial time. In the next section we provide some of the characteristics of the optimal solution, and provide a water-filling interpretation.

III. Optimal Transmission Strategy

For the optimization problem in (7), the following Lagrangian function is defined with the Lagrangian multipliers $\lambda_i \geq 0$, $\mu_i \geq 0$ and $\eta_i \geq 0$.

$$\mathcal{L} = \sum_{i}^{N} \tau_i \frac{e^{r_i} - 1}{h_i} + \sum_{j=1}^{N} \lambda_j \left(\sum_{i=1}^{j} \tau_i (d_i - r_i) \right) + \sum_{j=1}^{N} \mu_j \left(\left(\sum_{i=1}^{j} \tau_i (r_i - d_i) \right) - B \right) - \sum_{j=1}^{N} \eta_j r_j. \quad (8)$$

Additional complementary slackness conditions are given as follows:

$$\lambda_j \left(\sum_{i=1}^{j} \tau_i (d_i - r_i) \right) = 0, \qquad \forall j, \quad (9)$$

$$\mu_j \left(\left(\sum_{i=1}^{j} \tau_i (r_i - d_i) \right) - B \right) = 0, \qquad \forall j, \quad (10)$$

$$\eta_j r_j = 0, \qquad \forall j. \quad (11)$$

The only difference between the two sets of slackness conditions is that the second one includes another constant (cache capacity B); therefore, when, for some j, one of these conditions is satisfied with its parameter (λ_j or μ_j) being positive, then the parameter of the other condition has to be zero to satisfy the slackness condition. In other words, the following equation always holds:

$$\lambda_j \mu_j = 0, \qquad \forall j = 1, \ldots, N. \quad (12)$$

We apply the KKT optimality conditions on the Lagrangian to obtain:

$$\frac{d\mathcal{L}}{dr_i} = \frac{1}{h_i} \tau_i e^{r_i} - \sum_{j=i}^{N} \lambda_j \tau_i + \sum_{j=i}^{N} \mu_j \tau_i - \eta_i = 0. \quad (13)$$

Then the optimal transmission rate is found as follows:

$$r_i = \left[\sigma_i - \log \frac{1}{h_i} \right]^+, \quad (14)$$

where $[x]^+$ is equal to x if $x \geq 0$, and 0 otherwise, and we have defined the *water level* in TS i as

$$\sigma_i \triangleq \log \left(\sum_{j=i}^{N} \lambda_j - \mu_j \right). \quad (15)$$

A. Infinite Cache Capacity

We first consider the special case of a user with an infinite cache capacity. When $B \to \infty$, from the second set of slackness conditions in (10), we have $\mu_j = 0$ for $j = 1, \ldots, N$,

$$\sigma_i = \log \left(\sum_{j=i}^{N} \lambda_j \right). \quad (16)$$

Since $\lambda_i \geq 0$, $\forall i$, it follows that $\sigma_1 \geq \cdots \geq \sigma_N$, that is, the water level is decreasing with time. Moreover, if part of the user's demand in TS i is transmitted in advance in TS $i-1$, and stored in the cache memory, this implies that the i-th condition in (3) is satisfied with strict inequality. From the slackness condition in (9) this means that $\lambda_i = 0$; which, from Eqn. (16), leads to the fact that $\sigma_i = \sigma_{i+1}$, i.e., the water level remains constant.

When the cache memory is infinite the optimum transmission rate r_i^* can be written in the following form

$$r_i^* = \begin{cases} \log \sigma_i - \log \frac{1}{h_i}, & \text{if } \log \sigma_i \geq \log \frac{1}{h_i} \\ 0, & \text{if } \log \sigma_i < \log \frac{1}{h_i}, \end{cases} \quad (17)$$

which has a water-filling interpretation.

Note that the water levels are decreasing over time. This is because the water can only flow backwards as the demands are required to be satisfied by their individual deadline; that is, the rates can only be allocated to preceding TSs, not to

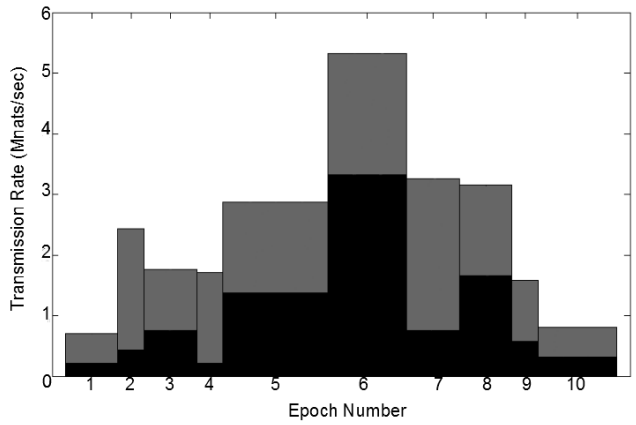


Fig. 2. Rate allocation over time without proactive caching. Each user request is satisfied within the timeslot it arrives.

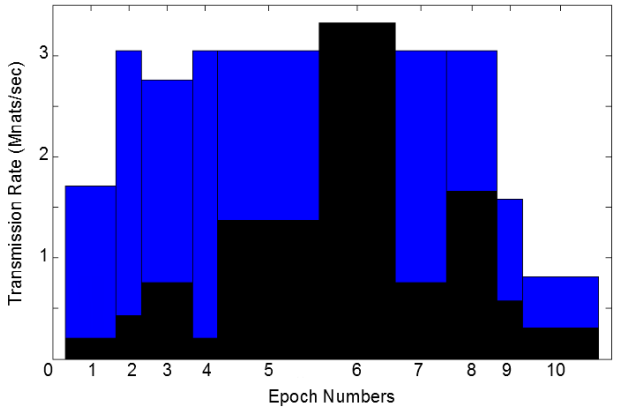


Fig. 3. Rate allocation for proactive caching with limited cache capacity. The optimal downloading rate values are obtained through backwards water-filling algorithm.

the future ones. As an example, we illustrate the optimal transmission strategy with infinite cache memory over three TSs. To obtain the optimal transmission policy, we first plot the inverse channel gains as in Fig. 1(a). Then, by considering the requests in the order they arrive, water-filling should be performed as illustrated in Fig. 1. The request in the first TS has to be satisfied within that TS, as seen in Fig. 1(b). For the second request, rate is allocated using the water-filling algorithm in the backwards direction as in Fig. 1(c). Since the channel is relatively poor in the second TS, some of the bits are downloaded in advance within the first TS, and r_1 is readjusted accordingly. The request in the third TS is satisfied within that TS. Thus, the rate in the first TS depends on the user's requests and channel conditions in the following TSs. By N iterations of the water-filling algorithm all the optimal rate values can be obtained. Since each request is satisfied by rate allocation over the previous TSs, the algorithm is called *sequential backwards water-filling*.

B. General Case

For the general case with finite cache memory and channel variations, we need to take into account the parameters μ_i. As the cache gets filled, μ_i increases; and therefore, r_i is limited. In other words, the cache memory introduces an upper bound on the rate at each TS, and this bound is imposed through the Lagrange multipliers μ_i.

The optimal solution for the general case is similar to the sequential backwards waterfilling solution in Section III-A; however, the amount of water that can be poured into a TS is now bounded by the cache capacity, since when the cache is full, increasing the rate would result in loss of data. As μ_i's are also non-negative, the water level σ_i is not necessarily decreasing, and can now increase from one TS to the next. This happens when the cache is full at a TS. We will illustrate this phenomenon through numerical examples in the next section.

IV. Numerical Results

In this section, some numerical results are presented in order to provide further insights into the optimal proactive caching algorithm introduced in the previous sections. As an input to our system, we fix the requested demand rates, channel gains and TS durations. For the rate-power function we consider $r(t) = W\log(1 + h(t)p(t))$, where $h(t)$ denotes the instantaneous channel signal-to-noise ratio, that is, the channel power gain divided by the noise power over the transmission bandwith W. We fix the bandwidth as $W = 1$ MHz.

We consider the scenario with ten TSs, i.e., $N = 10$. Demand rates for these 10 TSs are chosen as $\mathbf{d}= [0.5, 2, 1, 1.5, 1.5, 2, 2.5, 1.5, 1, 0.5]$ Mnats per second; the channel SNRs are $\mathbf{h}= [0.75, 0.55, 0.35, 0.75, 0.15, 0.01, 0.35, 0.1, 0.45, 0.65]$; and finally the TS durations are $\boldsymbol{\tau}= [2, 1, 2, 1, 4, 3, 2, 2, 1, 3]$ seconds. Under this model the channel changes relatively slowly, and the user demand profile has a time-scale similar to channel variations. Note that our model is sufficiently general to study different time scales for the channel and the demand rates. We set the cache memory size to $B = 2$ Mnats.

We first consider reactive resource allocation; that is, the base station does not track the user's future demands or channel conditions, and the user demand within each TS is satisfied at exactly the requested data rate. This would result in $r_i = d_i, \forall i$, and the corresponding rate allocation is illustrated in Fig. 2. Note that the cache memory is not utilized in the reactive scenario. The total energy requirement for reactive transmission is found to be 2173 J. On the other hand, when proactive caching is utilized, and the sequential backwards water-filling algorithm is employed, the optimal rate allocation is as given in Fig. 3. The total energy consumption drops significantly, to 323 J with proactive caching.

In Fig. 2 and Fig. 3 the black rectangles at the bottom correspond to channel states, and the portions above the black rectangles correspond to the optimal rate values for the TSs.

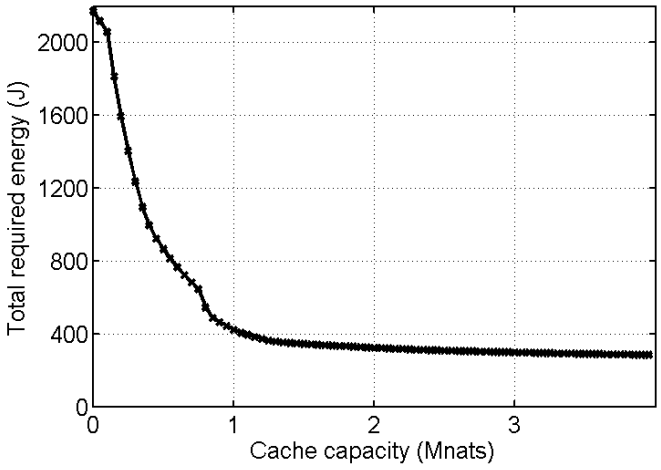


Fig. 4. Required total transmission energy vs. the available cache capacity at the user.

Note that all the demands are satisfied and the total energy requirement is minimized for each transmission strategy. In Fig. 3 the water levels are equalized to the extend the water-filling direction and the cache memory constraints allow. For instance, the data requested in TS 6 is downloaded in advance, and no transmission occurs in this TS, which has the worst channel gain. We also observe that the water level increases from TS 1 to TS 2 even though the channel is worse in the latter. This is because the cache capacity is already filled in TS 1, and no more data can be stored.

We also investigate the impact of the cache capacity on the required total energy. In Fig. 4, we study the same demand vector over time for the exact same channel conditions with increasing cache capacity from $B = 0$ to $B = 4$ Mnats. As expected, as the cache memory gets larger; first, the total energy requirement drops significantly thanks to the degree-of-freedom provided by the cache memory. However, after a certain point, increase in the cache capacity does not improve the energy performance any more. This is due to the fact that, once all the demands are satisfied at the best TS, there is no gain from an increase in the cache capacity.

V. Conclusions

We have studied proactive caching of content directly at a user device in order to minimize the total energy consumption. We have considered time-varying channel conditions, which may be caused due to mobility of the user or changing traffic conditions, and assumed that the channel conditions and user's requested download rates are known in advance over a certain period of time. Under the requirement that all the user's requests need to be satisfied, we have minimized the total energy requirement by proactively downloading the requested data when the channel is in a better state, and over a longer period of time.

The user has a local cache memory with finite capacity to store the proactively downloaded data. We have shown that the optimal transmission schedule can be formulated as a convex optimization problem, and the solution has suggested that a sequential backwards water-filling algorithm can be used to optimally download the content over time. Our simulation results consistently show that the proposed water-filling algorithm brings about significant energy gains compared to the conventional reactive data download, typically used in practical systems today. We have also identified how the energy requirement decays with the increasing cache capacity. Our results indicate that there is a lower bound for the total energy requirement, which can be achieved with a relatively low cache capacity.

Ongoing and future work [7], [8] embraces the extension of the proposed scenario to those with more than one user. In such a scenario broadcasting and device-to-device transmissions can be exploited in addition to proactive caching to further reduce the energy requirement of the system.

References

[1] Cisco visual networking index: Forecast and methodology, 2011-2016.

[2] H. Abou-zeid, H. Hassanein, and S. Valentin. Energy-efficient adaptive video transmission: Exploiting rate predictions in wireless networks. *IEEE T. Vehicular Technology 63*, Mar. 2014.

[3] E. Bastug, M. Bennis, and M. Debbah. Living on the edge: The role of proactive caching in 5g wireless networks. *IEEE Communications Magazine*, 52(8):82–89, Aug. 2014.

[4] P. Blasco and D. Gündüz. Learning-based optimization of cache content in a small cell base station. In *IEEE Int'l Conf. on Communications (ICC)*, Sydney, Australia, Jun. 2014.

[5] M. Dräxler, J. Blobel, P. Dreimann, S. Valentin, and H. Karl. Anticipatory buffer control and quality selection for wireless video streaming. *arXiv:1309.5491*, Sep. 2013.

[6] N. Golrezai, A. F. Molisch, A. G. Dimakis, and G. Caire. Femtocaching and device-to-device collaboration: A new architecture for wireless video distribution. *IEEE Communications Mag.*, 51(4):142–149, Apr. 2013.

[7] M. Gregori, J. Gomez-Vilardebo, J. Matamoros, and D. Gündüz. Wireless content caching for small cell and d2d networks. *submitted*, Apr., 2015.

[8] M. Gregori, J. Gomez-Vilardebo, J. Matamoros, and D. Gündüz. Joint transmission and caching policy design for energy minimization in the wireless backhaul link. In *Proc. IEEE Intl Symp on Inform Theory (ISIT)*, Hong Kong, China, Jun. 2015.

[9] Y. Huang and S. Mao. Downlink power control for multi-user VBR video streaming in cellular networks. *IEEE Trans.s Multimedia*, 15(8):2137–2148, Dec. 2013.

[10] F. Malandrino, M. Kurant, A. Markopoulou, C. Westphal, and U. C. Kozat. Proactive seeding for information cascades in cellular networks. In *Proc. IEEE INFOCOM*, Mar. 2012.

[11] S. Sadr and S. Valentin. Anticipatory buffer control and resource allocation for wireless video streaming. *arXiv:1304.3056v1*, Apr. 2013.

[12] J. D. Salehi, Z.-L. Zhang, J. F. Kurose, and D. Towsley. Supporting stored video: Reducing rate variability and end-to-end resource requirements through optimal smoothing. *IEEE/ACM Trans. Networking*, 6(6):397–410, Aug. 1998.

[13] V. A. Siris, X. Vasilakos, and G. C. Polyzos. Efficient proactive caching for supporting seamless mobility. *arXiv:1404.4754*, Apr. 2014.

[14] M. A. Zafer and E. Modiano. A calculus approach to energy-efficient data transmission with quality-of-service constraints. *IEEE/ACM Trans. Networking*, 17(3):898–911, Jun. 2009.