# QoE-Driven Resource Allocation for DASH over OFDMA Networks

Kefan Xiao, Shiwen Mao, and Jitendra K. Tugnait
Department of Electrical and Computer Engineering, Auburn University, Auburn, AL 36849-5201
Email: kzx0002@tigermail.auburn.edu, smao@ieee.org, tugnajk@eng.auburn.edu

***Abstract*—In this paper, we study the problem of video delivery over Orthogonal Frequency Division Multiple Access (OFDMA) networks using the Dynamic Adaptive Streaming over HTTP (DASH) framework. The goal is to integrate these two principal technologies to enable effective wireless video delivery. Based on a comprehensive QoE model, we develop a formulation to maximize the user QoE with joint OFDM resource allocation and DASH rate adaptation. The formulated problem is decomposed into a BS resource allocation problem and a user rate adaptation problem, which are then solved with effective algorithms. Our simulation study validates the efficacy of the proposed scheme.**

## I. Introduction

With the dramatic advances in wireless communications and networking, there has been an unprecedented increasing demand for mobile data service. According to a recent study by Cisco, globally, mobile data traffic will increase by 13-fold from 2012 to 2017. Traffic from wireless and mobile devices will exceed that from wired devices by 2016. In addition, video traffic has accounted for up to $64\%$ of the entire Internet traffic now, and mobile video, including all video data that travels over 2G, 3G, or 4G networks, is predicted to grow at a compound annual growth rate of 90% from 2012 to 2017. This trend is driven by the compelling need for ubiquitous access to data and video content for mobile users. The considerable mobile video data will pose great challenges to the capacity of existing wireless networks and strongly influence the design of future, i.e., 5G, wireless networks.

In this paper, we study the challenging problem of Dynamic Adaptive Streaming over HTTP (DASH) based video delivery over Orthogonal Frequency Division Multiple Access (OFDMA) networks. DASH has been adopted by many Internet video companies including Youtube and Netflix for its benefits of flexible adaptation to network congestion levels and quality of experience (QoE) provisioning. In DASH, video content is encoded into multiple versions with different bit rates and quality. Further, each version is partitioned into multiple non-overlapping chunks. A user uses HTTP/TCP to download the video chunk by chunk. Instead of directly controlling the transmission rate of video at the server, DASH provides users with high QoE through adaptively choosing a proper version (i.e., a proper bit rate) for different chunks, in reaction to network dynamics. QoE prediction in DASH is an interesting and important study area. Several recent works [1], [2] indicate that the DASH QoE model should be different from the traditional ones, where additional factors such as rebuffering events and initialization delay should be taken into account.

Orthogonal frequency division mutiplexing (OFDM) has been the core technique for many broadband wireless networks. It has been adopted in major standards such as 3GPP-Long Term Evolution (LTE), Wi-Fi, and will continue to present in future $5$th generation wireless networks. With OFDM, a spectrum is partitioned into multiple subcarriers. An important problem in OFDM networks is how to allocate *resource blocks*, in the form of a combination of time and frequency/subcarrier, as well as transmit power to multiple users, to achieve high spectral efficiency and guarantee user's quality of service (QoS) and QoE. There have been considerable existing works on resource allocation in OFDM networks. In [3], [4], the authors propose rate-adaptive (RA) schemes to maximize the throughput of users under transmit power budget constraints. In [5], energy efficiency is the primary goal for resource allocation in the downlink of an OFDM network, where proportional rates are guaranteed for mobile users.

We investigate the problem of DASH-based video delivery in OFDM networks, aiming to integrate these two principal technologies to enable effective wireless video delivery. In particular, we consider the downlink of an OFDM network, where the BS transmits multiple video streams to mobile users. The videos are encoded in the DASH format, i.e., multiple versions, while each partitioned into multiple chunks. The multiple video sessions share the downlink capacity. The BS dynamically allocates the downlink resource, in terms of time, subcarrier, and transmit power, to the mobile users, based on feedback on channel state information (CSI) and users' playout buffer occupancy. After downloading a chunk, each user also dynamically adjusts its data rate for the next chunk, based on its buffer level and CSI. The overall goal is to maximize the user QoE through joint BS and client side optimization.

Compared to the prior work on DASH (e.g., [6]), we consider a more realistic wireless network model and jointly optimize the operation of the underlying wireless network and the DASH system. We first consider a comprehensive QoE model that incorporates several factors including average and variance of the video quality, rebuffering ratio, and startup delay. We then develop a global offline formulation to maximize the sum QoE of all users through resource allocation and DASH rate adaptation. To avoid the use of future information, we break down the problem into a *BS resource allocation problem*, where the BS optimizes a weighted sum of user

978-1-5090-1328-9/16/$31.00 ©2016 IEEE

rates by allocating time, subcarrier, and transmit power, and a *user rate adaptation problem*, where each user chooses a data rate for the next chunk to maximize its own QoE. We develop effective solutions to both problems based on an analysis of their convexity property. The proposed scheme is validated with Matlab simulations, and is shown to outperform a benchmark scheme with considerable margins.

The remainder of this paper is organized as follows. We review related work in Section II. The system model and the problem formulation are presented in Sections III and IV, respectively. The solution algorithms are presented in Section V and evaluated in Section VI. Section VII concludes this paper.

## II. Related Work

This work is closely related to the prior works on resource allocation in OFDM networks and rate adaptation in DASH framework. Resource allocation in OFDM networks has drawn significant interests in the past decade [3]–[5]. See [7] for a comprehensive survey. Many prior works present general studies of energy efficient or system throughput maximization. In [8], the authors provided an analysis and solution on queue management, resource allocation, and subcarrier assignment in an OFDM network for transmission of MPEG-4 video. The two prior works [9], [10] study cross-layer optimization in OFDM networks. A more recent work [11] presents a scheme for the resource allocation and scheduling at the BS for Scalable Video Coding (SVC) video. One specific issue with the algorithm is that it only focuses on the BS but ignores the user playout states.

The DASH framework has been widely used for distribution of stored video in the Internet. There have been some interesting work in the literature focusing on DASH rate adaption at client side [12]–[14]. Generally, the existing approaches can be classified into three types: throughput estimation based, buffer occupancy based, and integrated schemes that use both throughput estimation and buffer occupancy. In [12], the authors present a scheme based on the classical proportional-integral-derivative (PID) control theory and throughput estimation to adjust video segment rate. A buffer based algorithm is developed in [13] to adjust the threshold for rate adaption. In [14], the authors consider both rate based and buffer based ideas and propose a model predictive control (MPC) based algorithm. One problem of the user side algorithm is that, the user can only passively adjust its rate to adapt to network dynamics, which may result in low QoE and waste of valuable resource, especially in a wireless network setting.

Perhaps the most related work to this paper is [6], which presents a joint resource allocation and rate adaption algorithm for DASH-based video delivery. It consists of both optimizations at the BS and video users. However, this work does not explicitly consider resource allocation details in OFDM networks, and uses a given downlink rate at the BS. Since the downlink rate depends on the CSI and power allocation of all users, it is actually a function of resource allocation in OFDM networks, rather than a given constant and an input for resource allocation. In addition, network congestion level is not explicitly considered in this existing work.

## III. System Model

### A. Network Model

We consider the edge of the network, i.e., the last wireless hop where a base station (BS) serves $U$ users with OFDM. We focus on the downlink transmissions of video data. Let the entire bandwidth be $B$, consisting of $S$ subcarriers, denoted as $s = 1, 2, ..., S$, and each with bandwidth $\nu = B/S$. Since $\nu$ is sufficiently small, each subcarrier only experience flat fading. Time is divided into equal length slots, denoted as $t = 1, 2, ...$, and each slot contains an integer number of OFDM symbols.

The channel gain of subcarrier $s$ to user $i$ at time slot $t$ is denoted as $g_{is}(t)$. Since multiple users can share subcarriers in each time slot, denote the non-overlapping time fraction of user $i$ on subcarrier $s$ in time slot $t$ as $0 \leq o_{is}(t) \leq 1$. Without loss of generality, assume the time slot is unit time, i.e., $\sum_{i=1}^{U} o_{is}(t) \leq 1$, for all $s$ and $t$. The average transmit power assigned to user $i$ on subcarrier $s$ in time slot $t$ is $p_{is}(t)$. The additive white Gaussian noise (AWGN) at the receiver has unit spectrum density. Then, the maximum rate user $i$ can achieve on subcarrier $s$ at time $t$ is

$$r_{is}(t) = \begin{cases} o_{is}(t)\nu \log_2 \left(1 + \frac{p_{is}(t) g_{is}^2(t)}{o_{is}(t)\nu}\right), & \text{if } o_{is}(t) > 0 \\ 0, & \text{otherwise.} \end{cases} \quad (1)$$

With channel state information (CSI), the BS allocates different power $p_{is}(t)$ and subcarrier time fraction $o_{is}(t)$ to user $i$ on subcarrier $s$ in each time slot $t$. The maximum rate user $i$ can achieve at time $t$ is $r_i(t) = \sum_{s=1}^{S} r_{is}(t)$. The overall energy budget of the BS is $P_b$.

### B. Streaming Video Model

Each video $i$ of $L_i$ seconds is partitioned into a set of $K_i$ consecutive segments, or chunks, each containing $l$ seconds of video content. We consider the case that each chunk has a constant bit rate, and each segment is encoded at a certain bit rate. Let $\Re$ denote the set of available video bit rates. After downloading chunk $k-1$, a user $i$ can choose a bit rate $R_i[k]$ for the next chunk $k$ from $\Re$. The size of a chunk is $R_i[k] \times l$.

We next describe the model on the user side. Intuitively, the higher the bit rate, the higher video quality a user will experience. Let function $q(R_i[k])$ represent the video quality with bit rate $R_i[k]$. We adopt the following form as in [15].

$$q(R_i[k]) = \alpha_i \log(R_i[k]) + \beta_i. \quad (2)$$

For different users, the parameters $\alpha_i$ and $\beta_i$ would be different. For example, mobile users are usually less sensitive to rate variation compared with desktop users, whose screen are normally bigger. Thus the function should be flatter for mobile users and stiffer for desktop users.

The video chunks are first downloaded into a playback buffer, which stores the video chunks that are downloaded but not viewed. Let $t_i[k]$ represent the time when chunk $k$ is downloaded to user $i$, and $B_i[k] \in [0, B_{i,max}]$ the buffer occupancy at time $t_i[k]$. Note that buffer occupancy

is represented in seconds, which means the total display time for the content in the buffer. Different users may have different buffer sizes $B_{i,max}$, which normally can store tens of seconds of video data.

The downloading time of segment $k$ is $R_i[k]l/C_i[k]$, where $C_i[k]$ is the average capacity assigned to user $i$ when downloading segment $k$. After chunk $(k-1)$ is fully downloaded, user $i$ will send a request `get` to the server for chunk $k$. The time between when the request is sent and when the chunk is ready to be downloaded at the BS is $t_i^P[k]$, which depends on several network parameters and network congestion level. It usually changes over time.

Such a discrete time system can be modeled as follows. The timeline of the operations and updates is presented in Fig. 1.

$$\begin{cases} B_i[k] = \min\left\{\max\left\{0, B_i[k-1] - \frac{R_i[k]l}{C_i[k]}\right\} + l, B_{i,max}\right\} \\ C_i[k] = \frac{\int_{t_i[k-1]+t_i^P[k]}^{t_i[k]} r_i(t)dt}{t_i[k]-t_i[k-1]} \\ t_i[k] = t_i[k-1] + \frac{R_i[k]l}{C_i[k]}. \end{cases} \tag{3}$$

Note that with the capacity definition $C_i[k]$ in (3), we have $\frac{R_i[k]l}{C_i[k]} = t_i[k] - t_i[k-1]$. Further, when $B_i[k-1] < \frac{R_i[k]l}{C_i[k]}$, the buffer will be empty, and a *rebuffering event* will occur and the rebuffering time is $\frac{R_i[k]l}{C_i[k]} - B_i[k-1]$. Rebuffering events significantly degrade the user QoE, which, therefore, should be avoided [1].

### C. Quality of Experience Model

There have been some QoE models proposed for DASH in the literature [1], [16]. In this paper, we introduce a general QoE model. First, the key factors that affect the user QoE are enumerated in the following. Our QoE model is then a weighted sum of these factors, while the weights are different for different users and application scenarios.

- *Average video quality* $m_i[K] = \frac{1}{K}\sum_{k=1}^{K} q(R_i[k])$: it represents the video quality level averaged over the entire video playback period [14].
- *Variance of video quality* $\text{Var}_i[K]$: it accounts for the quality variation from chunk to chunk, given by $\text{Var}_i[K] = \frac{1}{K}\sum_{k=1}^{K}(q(R_i[k]) - m_i[K])^2$. Compared to average video quality, the variance has a bigger impact on the QoE [14].
- *Rebuffering ratio* $\text{Reb}_i[K]$: for chunk $k$, if the downloading time is larger than the buffer occupancy, i.e., $R_i[k]l/C_i[k] > B_i[k-1]$, rebuffering will occur and the rebuffering time is $R_i[k]l/C_i[k] - B_i[k-1]$. The rebuffering ratio is defined as the total rebuffering time over the total video duration $L_i$, i.e.,

$$\text{Reb}_i[K] = \frac{1}{L_i}\sum_{k=1}^{K}\max\left\{0, \frac{R_i[k]l}{C_i[k]} - B_i[k-1]\right\}. \tag{4}$$

  This factor affects the QoE even more significantly than the variance [2].
- *Startup delay* $T_i^s$: it represents the time between user requests a video and the playback begins. Normally, a certain length of buffer occupancy need to be accumulated before playback starts [14]. It depends on the transmission rate and how the video is encoded.

We define user QoE as a weighted sum of all these factors.[1]

$$\begin{aligned} \text{QoE}_i[K] &= m_i[K] - \theta\text{Var}_i[K] - \lambda\text{Reb}_i[K] - \eta T_i^s \qquad (5) \\ &= \frac{1}{K}\sum_{k=1}^{K} q(R_i[k]) - \frac{\theta}{K}\sum_{k=1}^{K}(q(R_i[k]) - m_i[K])^2 - \\ &\quad \lambda\sum_{k=1}^{K}\frac{1}{L_i}\max\left\{0, \frac{R_i[k]l}{C_i[k]} - B_i[k-1]\right\} - \eta T_i^s. \end{aligned}$$

This model can be flexibly tuned for different users and application scenarios with different parameter sets $\{\theta, \lambda, \eta\}$. For example, mobile users are usually less tolerant to start-up delay than desktop users. So mobile users can adopt a larger $\eta$ value. Further, if the viewing process is emphasized instead of the startup phase, the stable QoE can be defined as

$$\text{QoE}_i^s[K] = m_i[K] - \theta\text{Var}_i[K] - \lambda\text{Reb}_i[K]. \tag{6}$$

## IV. Problem Formulation

With the above model, the BS and users operate on different timescales. Specifically, the BS adapts to the variation of network state (e.g., CSI and congestion) in every time slot $t$, while users choose the data rate for the next chunk after the previous one has been fully downloaded (i.e., at every $t_i[k]$).

### A. The Global Optimization Problem

With the models described in Section III, we now formulate the global optimization problem. Resource allocation at the BS involves variables $(p_{is}(t), o_{is}(t))$ in every time slot $t$; at the user side, the data rate $R_i[k]$ for each video chunk will be determined at each time $t_i[k-1]$. We formulate the optimization problem as

$$\max_{\{p_{is}(t), o_{is}(t), R_i[k]\}} \Lambda(U) = \sum_{i=1}^{U}\text{QoE}_i[K_i] \tag{7}$$

$$\text{subject to: } \sum_{i=1}^{U} o_{is}(t) \le 1,\ \forall s, t \tag{8}$$

$$\sum_{i=1}^{U}\sum_{s=1}^{S} p_{is}(t) \le P_b,\ \forall t \tag{9}$$

$$R_i[k] \in \Re_i,\ B_i[k] \in [0, B_{i,max}],\ \forall i, k. \tag{10}$$

Note that this formulation is based on the knowledge of the entire playback process. Such an *offline* optimization may not be practical in a realistic environment (due to the lack of future information). For a practical solution, we next develop a simpler *online algorithm*, which decouples the BS and user optimizations and does not require future information.

[1]For DASH, usually $B_{i,max}$ is set to be sufficiently large to avoid buffer overflow. Even when there is buffer overflow, there will be retransmission of the lost video data since HTTP/TCP is used. Therefore in this paper, we ignore the impact of buffer overflow on QoE for simplicity. In addition, the formulation in Section IV-B1 will make sure that a user gets no downlink rate if its buffer is close to full, thus avoiding buffer overflow events.

### B. Online Optimization Formulation

From the above formulation, the BS determines the transmission rate of each user through resource allocation based on the CSI at the beginning of each time slot, which next affects the playout buffer occupancy. The buffer occupancy depends on the transmission rate of the wireless link (i.e., the input), and the playout process (i.e., the output). An increased buffer level indicates that the video quality can be enhanced by choosing a larger $R_i[k]$ (since the buffer occupancy is in unit of time, not bits). In addition, the user with a higher risk of rebuffering should be allocated more resource to keep its buffer from underflow. We present the online optimization in the following, which consists of two parts, i.e., optimization at the BS side and at the client side.

*1) Base Station Side Optimization :* The BS allocates resource to users in every time slot $t$. The BS obtains the $B_{i,max}$ value when the video session is initiated, and receives feedback from each user $i$ on its buffer occupancy $B_i[t]$ every time when user $i$ requests for the next chunk $(k+1)$ at $t_i[k]$. Over time slots, $B_i[t]$ evolves as

$$B_i[t] = \begin{cases} \max\{0, B_i[t-1]-1\}, & t_i[k-1] \le t < t_i[k] \\ \max\{0, B_i[t-1]-1\}+l, & t = t_i[k], \end{cases} \quad k = 1, 2, ..., K_i. \tag{11}$$

With the buffer information, we define the *weight of resource allocation* at the BS as

$$\omega_i(t) = \frac{\alpha_i(t)}{\sum_{i=1}^{U} \alpha_i(t)}, \text{ for } i = 1, 2, ...U, \tag{12}$$

where $\alpha_i(t) = \log\left\{\frac{B_{i,max}}{B_i[t]+\eta}\right\}$ for all $t$, and $\eta$ is a small constant. We then formulate the BS problem to maximize the weighted sum of the downlink rates of all users as follows.

$$\max_{\{p_{is}(t), o_{is}(t)\}} \Phi(p_{is}(t), o_{is}(t)) = \sum_{i=1}^{U} w_i(t) \sum_{s=1}^{S} r_{is}(t) \tag{13}$$

$$\text{subject to: } \sum_{i=1}^{U} o_{is}(t) \le 1, \ \forall s, t \tag{14}$$

$$\sum_{i=1}^{U} \sum_{s=1}^{S} p_{is}(t) \le P_b, \ \forall t. \tag{15}$$

It can be seen that when $B_i[t]$ reaches $B_{i,max} - \eta$, $\alpha_i(t)$ will be 0, and thus the weight for this user will be 0. The BS will not allocate any resource to this user, and its buffer will only decrease. Thus we can prevent buffer overflow with this mechanism. On the other hand, a user with a small buffer size will have a large weight and get a relatively high downlink rate, thus preventing rebuffering at this user.

*2) Client Side Optimization:* The user side adaptation will be executed every time after a chunk is fully downloaded. Although the data rate in the downloading process is hard to predict, it's reasonable to assume the average downloading rate to be the same in two consecutive downloading periods. The average downloading capacity is determined by two factors: the downloading startup delay for chunk $k$, $t_i^P[k]$, and the assigned transmission rate when the user starts to download chunk $k$ (see Fig. 1).

At $t_i[k-1]$, user $i$ aims to maximize its QoE by choosing $R_i[k]$, based on $B_i[k-1]$ and history information. We formulate the user $i$ side problem as

$$\max_{R_i[k]} \text{QoE}_i[k] \tag{16}$$

$$\text{subject to: } R_i[k] \in \Re_i, \tag{17}$$

$$B_i[k] \in [0, B_{i,max}], \tag{18}$$

where $\text{QoE}_i[k]$ is defined as

$$\text{QoE}_i[k] = q(R_i[k]) - \theta(q(R_i[k]) - m_i[k])^2 - \frac{\lambda}{L_i} \max\left\{0, \frac{R_i[k]l}{C_i[k]} - B_i[k-1]\right\}, \tag{19}$$

where $m_i[k]$ is the mean video quality for the first $k$ video chunks, and $C_i[k]$ can be approximated by $C_i[k-1]$.

## V. Solution Algorithms and Analysis

The solution algorithm to the online optimization problems consists of three parts: (i) BS optimization (BSOP), (ii) user data rate adaption (UDRA), and (iii) parameter update (PUD), which are presented in this section.

### A. BS Optimization (BSOP)

The BS allocates the OFDM network resource $(p_{is}, o_{is})$, $i = 1, 2, ..., U$, $s = 1, 2, ..., S$, to users by solving the BSOP problem in every time slot $t$, given the constraints on power budget and time fractions based on the CSI each time slot. It requires $B_i[t]$ for all users for computing the weights, as given in (11) and (12).

**Theorem 1.** *Problem* (13) *is a convex optimization problem.*

The proof is omitted due to lack of space. We can apply a convex optimization solver, e.g., the Lagrange dual approach, to effectively solve the BS resource allocation problem.

### B. User Data Rate Adaption (UDRA)

At the client side, the distributed controller adapts video data rate $R_i[k]$ for each chunk $k$ after downloading chunk $(k-1)$. The client optimization problem (16) can be solved with an exhaustive search approach when the search space (i.e., the number of rates) is not large. Alternatively, we analyze the problem and provide a faster solution approach in this section.

**Theorem 2.** *The objective function* $\text{QoE}_i[k]$ *of problem* (16) *is neither convex nor concave.*

*Proof:* Consider function $f(R_i) = q(R_i) - \theta(q(R_i) - m_i)^2$, where $q(R_i) = a + b\ln(R_i)$. The first and second derivatives of this function are

$$\begin{cases} \frac{\partial f}{\partial R_i} = \frac{b}{R_i}\left(1 - 2\theta(q(R_i) - m_i)\right) \\ \frac{\partial^2 f}{\partial R_i^2} = -\frac{b}{R_i^2}\left(1 + 2\theta - 2\theta(q(R_i) - m_i)\right). \end{cases} \tag{20}$$

Setting the second derivative to be 0, we have

$$\begin{cases} \frac{\partial^2 f}{\partial R_i^2} \leq 0, \text{ when } q \in \left(0, \frac{1}{2\theta} + 1 + m_i\right] \\ \frac{\partial^2 f}{\partial R_i^2} > 0, \text{ when } q \in \left(\frac{1}{2\theta} + 1 + m_i, \infty\right). \end{cases} \tag{21}$$

Thus, the sum of the first two terms in (19) is neither convex nor concave. In addition, the third term in (19) is a piece-wise-linear continuous function (with two pieces). Since adding a linear function conserves the convexity or concavity of the original function, the sum of the three terms is still neither convex nor concave. ■

Although the proof of Theorem 2 does not support the convexity of the objective function, it does provide useful insights that lead to the following method for searching for the maximizer.

From (21), when $q \in \left(0, \frac{1}{2\theta} + 1 + m_i\right]$, the combination of the first two terms in (19) is a concave function. For $q \in \left(\frac{1}{2\theta} + 1 + m_i, \infty\right)$, the first derivative $\frac{\partial f}{\partial R_i} = \frac{b}{R_i}\left(1 - 2\theta(q(R_i) - m_i)\right) < 0$. Thus, the maximizer must be either in the concave range of function $f()$ or at the intersection point where $q(R_i[k]) = \frac{1}{2\theta} + 1 + m_i$, from which we can solve for the intersection point as

$$\bar{R}_i[k] = \exp\left(\frac{1}{b}\left(\frac{1}{2\theta} + 1 + m_i - a\right)\right). \tag{22}$$

On the other hand, the two linear pieces of the third term in (19) intersects at

$$\tilde{R}_i[k] = \frac{1}{l} C_i[k] B_i[k-1]. \tag{23}$$

These two values $\{\bar{R}_i[k], \tilde{R}_i[k]\}$ partition the search space for $R_i[k]$ into three regions. We consider the following two cases when search for the optimal solution in these three regions.

*a)* $\bar{R}_i[k] \geq \tilde{R}_i[k]$*:* In this case, the convex part of $f()$ is not affected by adding the third term in (19), since the third term remains 0 for the entire region where $f()$ is convex. Thus, the optimum can still be calculated by comparing the $\text{QoE}_i[k]$ at $\bar{R}_i[k]$ and the optimum $\text{QoE}_i[k]$ in the concave region. The latter can be found with a standard convex problem solver.

*b)* $\bar{R}_i[k] < \tilde{R}_i[k]$*:* This is a more complicated scenario. Still, the concavity of $f()$ will not be affected by adding the third term. Thus the optimum of the concave region can be found with a convex solver. Further, the third term in (19) is piece-wise linear with two pieces. Thus, the convexity property of (19) in range $(\bar{R}_i[k], \tilde{R}_i[k])$ and that in range $(\tilde{R}_i[k], \infty)$ will be maintained after adding the third term, although we cannot claim that the function is convex in the range.

Therefore, we search for the optimum in the concave part first, which can be solved by a convex solver. We then evaluate $\text{QoE}_i[k]$ at both $\bar{R}_i[k]$ and $\tilde{R}_i[k]$. Eventually we select the $R_i^*[k]$ that achieves the largest $\text{QoE}_i[k]$ among the three. Note that since this is an integer programming problem, we finally *round-up* a non-integer solution to an integer one. This way, we can greatly narrow down the search space and reduce the computational complexity at the user side.

### C. Parameter Update (PUD)

In the solution shown in Sections V-A and V-B, the system parameters need to be updated for every period of operation. At the BS side, the user buffer occupancy $B_i[t]$ will be updated as follows (also see (11)).

- When a user finishes downloading a chunk and sends a request to the server for the next chunk: the user will notify the BS its current buffer occupancy $B_i[t]$.
- Between user updates: $B_i[t] = \max\{0, B_i[t-1] - 1\}$, since the BS operates in unit time.

From the user perspective, the parameters that need to be updated are the mean quality $m_i[k]$, the average downloading capacity $C_i[k]$, and buffer occupancy $B_i[k]$. The updating law is given as follows.

$$\begin{cases} m_i[k] = \frac{k-1}{k} m_i[k-1] + \frac{1}{k} q(R_i[k]) \\ C_i[k] = \frac{R_i[k] l}{t_i[k] - t_i[k-1]} \\ B_i[k] = \min\{\max\{0, B_i[k-1] - t_i[k] + \\ \qquad\qquad t_i[k-1]\} + l, B_{i,max}\}. \end{cases} \tag{24}$$

## VI. Simulation Study

### A. Simulation Scenario and Algorithm Configuration

We evaluate the performance of the proposed algorithm with Matlab simulations in this section. The OFDM downlink transmissions are simulated. The total bandwidth is 20 MHz and consists of 64 subcarriers, each with 320 KHz bandwidth. The energy budget $P_b$ is set to 1 W. The channel gain for unit noise energy is assumed to be a random process (Rayleigh channel) with expectation 5 dB and remains constant in each time slot of 5 ms. The BS can acquire the CSI of each user by the pilot training procedure. The additive white Gaussian nose has a power density of $-60$ dB W/Hz.

There are $U = [2, 3, ..., 14]$ users in the network. The length of each video segment is 1 s. Each video is coded into five levels of rates, i.e., $\Re$ = [100 kbps, 300 kbps, 500 kbps, 900 kbps, 1500 kbps, 2000 kbps], which is consistent with the requirement of 240p, 360p, 480p, 720p,[2] and 1080p for general genre [17]. Each video lasts for 5 minutes. For the QoE model, we use constant $\alpha_i$ and $\beta_i$ for all users. Further, the download startup time $t_i^P[k]$ is assumed to follow an ergodic exponential distribution with expectation 50 ms. The quality model $\alpha_i$ and $\beta_i$ are fitted with the data from [2] and assumed to be the same for all the users. We set $\theta = 0.2$, $\lambda = 300$, and $\eta = 20$ according to [14].

For comparison purpose, we choose the proportionally fair network resource allocation with water-filling algorithm (PFWF) scheme as a benchmark [7], [18]. Video rate adaption based on rate matching (RM) are adopted at the BS and the user side. The main idea of RM is to choose a video rate to match the average throughput of the network [6].

### B. Simulation Results and Discussions

The resulting user QoEs achieved by the proposed scheme and benchmark are presented in Fig. 2. With the increasing

[2]Note that 900 kbps and 1500 kbps are the rates for 720p.

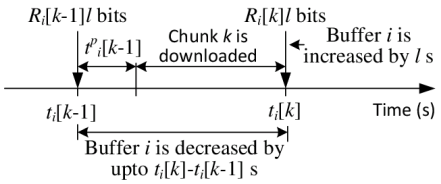


Fig. 1. Timeline of the discrete time DASH system.

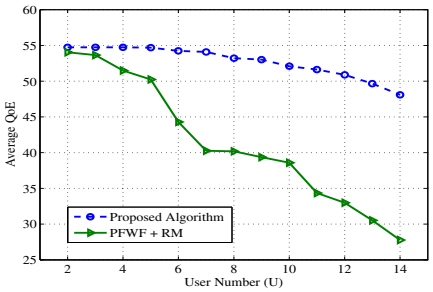


Fig. 2. Average QoE of the users.

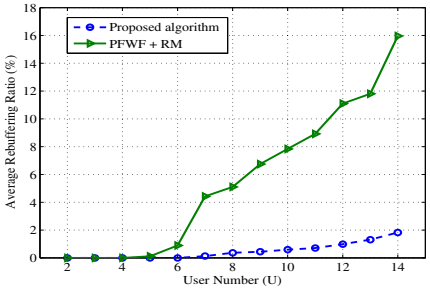


Fig. 3. Average rebuffering time ratios of the users.

number of video users, both QoE results are decreasing as expected. This is because the available resource for each user, on average, is decreasing as more users share the downlink OFDM link. The benchmark algorithm curve decreases at a much faster rate since the shortage on resource leads to more rebuffering events at the playout buffers, which greatly degrade the user QoE. The proposed algorithm, on the other hand, can handle this situation well, and can achieve an acceptable QoE value even under severe resource shortage. If we choose 50 as the threshold for an acceptable QoE value, the proposed scheme can support 12 users, while the benchmark scheme can only support 5 users.

In Fig. 3, the average rebuffering ratios of both schemes are plotted. It can be clearly seen that rebuffering events occur more frequently as more users become active, due to the shortage on downlink transmission resource (power and spectrum). However, the proposed algorithm can properly balance the buffer occupancy among all users and achieve high fairness among them. The benchmark algorithm curve quickly degrades as the user number is increased. The large rebuffering time ratio also explains why the average user QoEs achieved by the benchmark algorithm goes down quickly as $U$ is increased in Fig. 2.

## VII. Conclusions

In this paper, we investigated the problem of resource allocation for delivering multiple videos using DASH over the downlink of an OFDM network. We first presented an offline formulation based on a novel QoE model. We then derived a more practical online formulation, and developed a distributed solution algorithm, which consisted of an resource allocation algorithm at the BS side and a rate selection algorithm at the user side. The proposed scheme was validated with simulations and was shown to outperform a benchmark scheme with considerable gain margins.

## Acknowledgment

We thank Drs. Chao Chen and Gustavo de Veciana for providing some of the video model data used in this paper. This work is supported in part by the US NSF under Grant CNS-0953513, and by the Wireless Engineering Research and Education Center (WEREC) at Auburn University.

## References

[1] A. Balachandran, V. Sekar, A. Akella, S. Seshan, I. Stoica, and H. Zhang, "Developing a predictive model of quality of experience for internet video," in *Proc. ACM SIGCOMM'13*, Hong Kong, China, Aug. 2013, pp. 339–350.

[2] C. Chen, L. K. Choi, G. de Veciana, C. Caramanis, R. W. Heath, and A. C. Bovik, "A dynamic system model of time-varying subjective quality of video streams over HTTP," in *Proc. IEEE ICASSP'13*, Vancouver, Canada, May 2013, pp. 3602–3606.

[3] Z. Mao and X. Wang, "Efficient optimal and suboptimal radio resource allocation in OFDMA system," *IEEE Trans. Wireless Commun.*, vol. 7, no. 2, pp. 440–445, Feb. 2008.

[4] X. Wang and G. B. Giannakis, "Resource allocation for wireless multiuser OFDM networks," *IEEE Trans. Infor. Theory*, vol. 57, no. 7, pp. 4359–4372, July 2011.

[5] Z. Ren, S. Chen, B. Hu, and W. Ma, "Energy-efficient resource allocation in downlink OFDM wireless systems with proportional rate constraints," *IEEE Trans. Veh. Technol.*, vol. 63, no. 5, pp. 2139–2150, Mar. 2014.

[6] V. Joseph and G. de Veciana, "NOVA: QoE-driven optimization of DASH-based video delivery in networks," in *Proc. IEEE INFOCOM'14*, Toronto, Canada, Apr./May 2014, pp. 82–90.

[7] F. Shams, G. Bacci, and M. Luise, "A survey on resource allocation techniques in OFDMA networks," *Computer Netw.*, vol. 65, pp. 129–150, June 2014.

[8] J. Gross, J. Klaue, H. Karl, and A. Wolisz, "Cross-layer optimization of OFDM transmission systems for MPEG-4 video streaming," *Computer Commun.*, vol. 27, no. 11, pp. 1044–1055, July 2004.

[9] Y. Andreopoulos, N. Mastronarde, and M. Van der Schaar, "Cross-layer optimized video streaming over wireless multihop mesh networks," *IEEE J. Sel. Areas Commun.*, vol. 24, no. 11, pp. 2104–2115, Nov. 2006.

[10] G. Song and Y. Li, "Utility-based resource allocation and scheduling in OFDM-based wireless broadband networks," *IEEE Commun.*, vol. 43, no. 12, pp. 127–134, Dec. 2005.

[11] X. Ji, J. Huang, M. Chiang, G. Lafruit, and F. Catthoor, "Scheduling and resource allocation for SVC streaming over OFDM downlink systems," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 19, no. 10, pp. 1549–1555, July 2009.

[12] G. Tian and Y. Liu, "Towards agile and smooth video adaptation in dynamic HTTP streaming," in *Proc. 8th Int. Conf. Emerging Netw. Experiments Technol.*, Nice, France, Dec. 2012, pp. 109–120.

[13] T.-Y. Huang, R. Johari, N. McKeown, M. Trunnell, and M. Watson, "A buffer-based approach to rate adaptation: Evidence from a large video streaming service," in *Proc. ACM SIGCOMM'14*, Chicago, IL, Aug. 2014, pp. 187–198.

[14] X. Yin, A. Jindal, V. Sekar, and B. Sinopoli, "A control-theoretic approach for dynamic adaptive video streaming over HTTP," in *Proc. ACM SIGCOMM'15*, London, UK, Aug. 2015, pp. 325–338.

[15] C. Chen, X. Zhu, G. de Veciana, A. C. Bovik, and R. W. Heath, "Rate adaptation and admission control for video transmission with subjective quality constraints," *IEEE J. Sel. Topics Sig. Process.*, vol. 9, no. 1, pp. 22–36, Feb. 2015.

[16] J. Jiang, V. Sekar, H. Milner, D. Shepherd, I. Stoica, and H. Zhang, "CFA: A practical prediction system for video QoE optimization," in *Proc. USENIX NSDI'16*, Santa Clara, CA, Mar. 2016, pp. 137–150.

[17] S. Lederer, C. Müller, and C. Timmerer, "Dynamic adaptive streaming over HTTP dataset," in *Proc. ACM Multimedia'12*, Nara, Japan, Oct./Nov. 2012, pp. 89–94.

[18] H. J. Kushner and P. A. Whiting, "Convergence of proportional-fair sharing algorithms under general conditions," *IEEE Trans. Wireless Commun.*, vol. 3, no. 4, pp. 1250–1259, July 2004.