APPLICATION-LAYER QoS FAIRNESS IN WIRELESS VIDEO SCHEDULING

Tanir Ozcelebi, M. Oguz Sunay, M. Reha Civanlar, A. Murat Tekalp

College of Engineering, Koç University, 34450, Istanbul, Turkey.
{tozcelebi, osunay, rcivanlar, mtekalp}@ku.edu.tr

ABSTRACT

In mobile video transmission systems, the initial delay for pre-fetching video at the client buffer needs to be short due to buffer limitations and application-layer user convenience. Therefore, an effective cross-layer wireless design is required that considers both physical and application layer aspects of such a system. We present a cross-layer optimized multi-user video adaptation and scheduling scheme for wireless video communication, where Quality-of-Service (QoS) fairness among users is provided while maximizing user convenience and video throughput. Application and physical layer aspects are jointly optimized using a Multi-Objective Optimization (MOO) framework that tries to schedule the user with the least remaining playback time and the highest video throughput (delivered video seconds per transmission slot) with maximum video quality. Experiments with the IS-856 (1xEV-DO) standard and ITU Pedestrian A and Vehicular B environments show the improvements over today’s schedulers in terms of QoS fairness and user utility.

Index terms- Mobile communication, optimization, quality assurance, scheduling, video.

1. INTRODUCTION

As opposed to wired communication systems, the mobile environment comes with exceptional challenges and most such practical systems do not guarantee either network or application Quality-of-Service (QoS) for multiple users. Despite their high efficiencies, the state-of-the-art video codec standards such as Advanced Video Coding (AVC/H.264) [1]-[2] and scalable wavelet video coding [3] are not enough for this purpose. In a wireless one-to-many video-on-demand streaming network, the target is to achieve optimal user scheduling and source coding such that the highest application layer QoS is provided for each user fairly. Intelligent system designs that consider multiple aspects of the communications network are needed to overcome such a difficult task.

In the Open Systems Interconnect (OSI) layered design, the interfaces between different layers are fixed and they do not consider the system constraints of different OSI layers and applications, which is a suboptimal approach. The improvements of 2.5 and 3G systems over the voice centric 2G systems are insufficient for packetized high data rate services like video streaming. Therefore, adaptive and efficient system resource sharing schemes with high-speed wireless packet data access are needed. This paper focuses on time-multiplexed systems with adaptive coding and modulation as opportunistic multiple access schemes [4] give maximum channel capacity in frequency flat fading channels. Generally, there is a tradeoff between application layer delay and throughput in wireless systems. The user scheduling algorithm needs to consider both layers so that it can satisfy delay and throughput requirements of a specific application. Higher channel throughput can be achieved by making use of time-varying characteristics of the wireless system if longer delays are allowed.

The state-of-the-art schedulers are maximum rate [4], first-in-first-out (FIFO), proportionally fair (PF) [5] and exponential [6] schedulers. The inflexibility and sub-optimality of these general purpose scheduling algorithms with independent layers approach would result in poor performance for wireless video systems. High video streaming performance can be achieved via joint optimization of application and physical layer parameters where optimization goals may also be varied according to the streaming application. A communication system design whose rules are defined by considering multiple layers of the OSI protocol stack is called a cross-layer design. There have been several attempts for cross-layer adaptive design of video streaming systems in the literature [7]-[10]. None of these schemes consider cross-layer joint optimization of source (video) coding and network resource sharing.

In this paper, a cross-layer multi-objective optimized video adaptation and user scheduling framework for video streaming over wireless communication systems with multiple users requesting video streams from a single server is presented. Here application layer conveniences (video quality) of individual users as well as the overall “video throughput” are simultaneously maximized targeting

1 This work is supported by European Commission within FP6 under Grant 511568 with the acronym 3DTV.
continuous video playback. Note that video throughput, which is defined as the amount of delivered video seconds per transmission second, is different from the channel capacity as a concept. However, in case of constant bitrate, non-adaptive video transmission, maximizing video throughput corresponds to maximizing channel capacity.

This paper is organized as follows: In Section 2, the system optimization criteria are described. In Section 3, MOO problem for cross-layer design of channel resource allocation and video source coding in wireless channels is formulated and the method used for jointly optimizing multiple objective functions is presented. In Section 4, experimental results are given. Finally, in Section 5, conclusions are drawn.

2. SYSTEM OPTIMIZATION CRITERIA

Mobile video streaming has three system criteria as mentioned in Section 1, namely application-layer QoS fairness, overall video throughput and average delivered video quality criteria, which are individually investigated in this section.

For on-demand video streaming, the use of application-layer QoS fairness (video quality and continuous playback) is more meaningful than network-layer QoS fairness such as average data throughput and service latency. Especially in low-capacity networks, application-layer QoS fairness is a determining factor for the overall system excellence, that protects users with constantly and relatively worse channel conditions from experiencing bad video service. Once the video playback at the clients has started, video service fairness requires continuous playback of the received content for all users, which depends on bandwidth limitations, variations and packet losses in the network.

Assume that the total number of users with streaming video request in the wireless system is $M$ and let $n$ ($1 \leq n < \infty$) denote the discrete time slot index. Let the remaining video playback time for user $i$ at time slot $n$ before its buffer gets empty in case user $i$ is never scheduled ever again be denoted by $\theta_i(n)$. In order to provide continuous playback, the scheduler needs to select the user $i$ with minimum $\theta_i(n)$ at each time slot.

The amount of video content delivered to each user is an important service quality parameter. This quantity, called “video throughput” in this paper, is video seconds conveyed per transmission second and needs to be maximized. Let the instantaneous available channel data rate at time $n$ be $\lambda_i(n)$. Let the average video throughput up to the $n^{th}$ time slot be denoted by $t_i(n)$. The cross-layer scheduler decides on the value and the destination of the channel throughput at any given time along with the source encoding rate, indicated by:

$$t_i(n) = \frac{n-1}{n} \cdot t_i(n-1) + \frac{1}{n} \sum_{k=1}^{M} a_i(n) \cdot t_i(n) \quad (1)$$

Here $t_i(n) = \lambda_i(n) / \mu_i(n)$ denotes the video throughput that can be transmitted to user $i$ at time $n$, $\mu_i(n)$ is the encoding rate of the $i^{th}$ video stream and $a_i(n)$ is a binary variable that takes the value “1” if the user $i$ is scheduled and “0” otherwise. In order to maximize the average video throughput, the scheduler selects the user $i$ and video stream $l$ with the highest video throughput, $t_i(n)$.

The transmitting side needs to adopt not only its channel transmission rate, but also its video encoding rate. The average video quality for each user, hence the encoding rate, needs to be maximized in order for the best user experience. In the proposed framework, Automatic Repeat reQuest (ARQ) technique is employed and lost data in the channel is retransmitted. Hence the video quality depends solely on the encoding strategy rather than the communication channel behavior.

3. PROBLEM FORMULATION AND SOLUTION

In this section, the cross-layer multi-objective optimization (MOO) problem for multiuser streaming is formulated. The proposed problem formulation provides optimal source bitrate control and channel user scheduling, where video throughput and video rate (quality) are jointly maximized in a so called “best compromise solution” with maximum video QoS fairness. The use of MOO technique is beneficial in terms of its ease in sensitivity analysis.

We assume that the quantized available channel data throughput, $\lambda_i$, and remaining video playback times, $\theta_i(n)$, for each client are known by the base station at each time slot by means of physical and application layer feedback. The objective function triplet to be optimized amongst users at time $n$ is $(t_i(n), \theta_i(n), \mu_i(n))$, where $t_i(n)$ is the effective video throughput, $\mu_i(n)$ is the video rate to be transmitted at time $n$ if user $i$ and video stream $l$ were selected. At time $n$, the proposed optimizer selects the user-video stream pair $(i,l)$ that yields the best performance in these three target functions. Ideally, the server side must schedule the user that experiences the best compromise between the shortest remaining playback time and the largest available video throughput enhancement, with the best video quality. Hence, our formulation for scheduling a user $i$ at time slot $n$ with bitstream $l$ is given by

$$\max_{i,l} (t_i(n)) \quad (2)$$

$$\min_{i} (\theta_i(n)) \quad (3)$$

$$\max_{i,l} (\mu_i(n)) \quad (4)$$

jointly subject to buffer constraints, i.e. $0 < B_i(n) < \text{BufferSize}$ for all $i$ where $\text{BufferSize}$ is the client receiving buffer size. The validity of this constraint is checked by the buffer overflow/underflow flags fed back to the server.

The values of $t_i(n), \theta_i(n)$ and $\mu_i(n)$ needs to be scaled to the ranges $[0,w_1], [0,w_2]$ and $[0,w_3]$, respectively, where
\( w_p \) is the importance weight for the \( p^{th} \) objective function. The determination of these weights can be done by the server side according to the user profile.

In the solution of an optimization problem with the objective/cost function set \( F = \{ f_1, f_2, \ldots, f_p \} \), a solution \( s^* \) is called globally Pareto-optimal (also non-dominated/non-inferior) if any one of the objective function values cannot be improved without degrading other objective values. Let us assume that the optimization problem in hand consists of \( P \) distinct and conflicting objective functions. Without loss of generality, let us assume that the problem in hand requires all the objective functions to be minimized. Then, there exists no other feasible solution \( s \) that satisfies

\[
\forall p \in \{1,...,P\} \quad f_p(s) \leq f_p(s^*),
\]

with at least one strict inequality. In the proposed framework, we determine the best compromise solution on a fairness basis among objective functions by rescaling their amounts to an interval \([0,w_p]\), \( w_p \) being the importance weight of the \( p^{th} \) objective function using the equation:

\[
f_{p,\text{scaled}}(n) = w_p \times \frac{f_p(n) - f_{\text{min}}(n)}{f_{\text{max}}(n) - f_{\text{min}}(n)} \quad (5)
\]

The video throughput enhancement, user remaining playback time and video rate values are normalized to the interval \([0,w_p]\), and a three-dimensional solution space is formed. Note that, ideally the optimizer would select higher video bitrates when the users’ remaining play times are high and lower bitrates when they are low. For this purpose, the importance weight of the 3\(^{rd} \) objective function for maximizing the video rate, \( w_3 \), can be dynamically changed at each time slot according to the average remaining playback time for all users, \( \bar{\theta}(n) \), i.e.

\[
w_3 = \bar{\theta}(n)/\theta_{\text{max}}. \quad \text{Here} \quad \theta_{\text{max}} \text{is the maximum possible remaining playback time, i.e. the ratio of the buffer size to the lowest available video encoding rate.}
\]

In MOO problems, an infeasible point that optimizes all of the objective functions individually is called the utopia point. The best compromise solution is found as the feasible point that is closest to the utopia point in the Euclidian-distance sense. The utopia point, \( U(n) \), on the three-dimensional scaled video throughput, remaining-play-time, video rate space is set as follows:

\[
U(n) = (\max(\theta_{i,\text{scaled}}(n)), \min(\theta_{i,\text{scaled}}(n)), \max(\theta_{i,\text{scaled}}(n))) \quad (6)
\]

Knowing the client preferences, the server side may prefer to skip the original optimal solution and offer different solutions. A detailed explanation of the MOO techniques used in the literature can be found in [11]-[12].

### 4. EXPERIMENTAL RESULTS

Adaptive coding and modulation are used to deliver different levels of services (data rates) to end users on CDMA/HDR [13] systems, and the IS-856 standard [14], also called 1xEvolution-Data-Optimized (1xEV-DO). An opportunistic multiple access scheme that assigns all transmission power to only one user at a time within time slots of length 1.667 ms is employed. The back-channel utilizes available data throughput levels experienced by mobile devices using 13 pre-quantized levels. Hence, the base station knows the maximum transmission bitrate that can be achieved for each user within a probability of bit-error range; and past channel statistics can be stored and accessed at the transmitting side for better system performance.

The physical layer simulations have been performed on the ITU Pedestrian A and Vehicular B test environments using Agilent’s Advanced Design System (ADS 2004A), where a 3-tier cellular layout is assumed. The system sampling rate is 1296Hz, which also matches the Data Rate Request Channel (DRC) update rate used in 1xEV-DO. Videos of 183 seconds total duration are streamed to a total of 32 users, who are repeatedly and randomly dropped into the wireless cell in a uniformly distributed manner. One period of these drops is chosen to be 1 second, which is equal to 600 slots. A user is not scheduled if his/her receiving buffer will overflow. Note that, we need additional application layer feedback to report start and pause time instances of playback to the server for accurate calculation of remaining video playback times of clients. A 1-bit flag is sufficient for this purpose, which has a negligible impact on the uplink channel since the signaling intervals are long, i.e. in the order of tens of seconds.

For the duration of video streaming to 32 users, each with a maximum receiving buffer size of 1000 kbits, we measure the average and worst case number of pauses per playback second, PN; and average and worst case wait times (pre-roll plus pause delays), \( T_w \). A network packet error rate of 1% is simulated using a uniform distribution and IP/UDP/CRTPT network packetization is assumed. ARQ’s are employed in case of packet losses. In [15] we showed that the MOO cross-layer scheduler provides significant improvements over the state-of-the-art schedulers. Here we show that the inclusion of video adaptivity gives even better results as in Table 1 with variable bitrate (VBR) coding. VBR encoding for videos to be streamed is achieved by switching amongst several pre-encoded bitstreams at 50, 60, 70 and 80 kbps and 12.5 frames-per-second (fps) for each video. Switching among different bitstreams is possible only at specific instants allowed by the used video codec. In our experiments, we used the AVC/H.264 codec with a GoP size of 12 frames with the following structure: IPPPPPPPPPPP. Therefore, switching is possible every 12\(^{th}\) frame. In Table 1, Avg. Tw, Max. Tw, Avg. PN and Max. PN correspond to average and worst case cumulative waiting times and number of pauses per playback second, respectively.

The buffer size constraint may cause a drop in the overall throughput, since we cannot serve the user whose
channel is the best if its buffer is already full. Therefore, the maximum rate scheduler’s channel throughput performance is not necessarily the best in our experiments. If we were to assume that the receiving buffer sizes of users are infinitely large, then the maximum rate scheduler would have the best channel capacity, as the theory implies. Optimal video rate adaptation scheme arranges the transmitted video bitrate according to the amount of remaining video time at users' buffers. At the same average video rate, this has not only improved the continuous playback, but also reduced the channel capacity degradation due to buffer constraint.

5. DISCUSSION AND CONCLUSIONS

In this paper, we proposed a cross-layer optimized multiuser video adaptation and scheduling scheme for video communication over packetized networks, where QoS fairness (continuous video play) among users is provided while maximizing video quality and video throughput. A joint multi-objective optimization (MOO) formulation and a solution methodology are provided. Our cross-layer optimized results using the IS-856 (1xEV-DO) standard and ITU Pedestrian A and Vehicular B environments demonstrate that the number of undesired pauses during playback and initial pre-roll delays are considerably reduced by this technique compared to the state-of-the-art schedulers at the same average video rate.

6. REFERENCES


### TABLE I

<table>
<thead>
<tr>
<th>Environment: ITU Pedestrian A</th>
<th>Avg. video rate: 60 kbps, Initial buffer: 6 video seconds, Buffer size: 1000 kbits</th>
</tr>
</thead>
<tbody>
<tr>
<td>Scheduler</td>
<td>Avg. Tw (sec)</td>
</tr>
<tr>
<td>MOO with rate adaptation</td>
<td>5.3024</td>
</tr>
<tr>
<td>MOO with CBR video</td>
<td>7.0929</td>
</tr>
<tr>
<td>Proportionally Fair</td>
<td>29.1139</td>
</tr>
<tr>
<td>Exponential</td>
<td>32.3292</td>
</tr>
<tr>
<td>Maximum Rate (C/I)</td>
<td>68.5008</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Environment: ITU Vehicular B</th>
<th>Avg. video rate: 50 kbps, Initial buffer: 6 video seconds, Buffer size: 1000 kbits</th>
</tr>
</thead>
<tbody>
<tr>
<td>Scheduler</td>
<td>Avg. Tw (sec)</td>
</tr>
<tr>
<td>MOO with rate adaptation</td>
<td>24.6716</td>
</tr>
<tr>
<td>MOO with CBR video</td>
<td>24.7248</td>
</tr>
<tr>
<td>Proportionally Fair</td>
<td>48.2845</td>
</tr>
<tr>
<td>Exponential</td>
<td>50.0746</td>
</tr>
<tr>
<td>Maximum Rate (C/I)</td>
<td>73.5168</td>
</tr>
</tbody>
</table>