QoE Matters More Than QoS: Why People Stop Watching Cat Videos

Hyunwoo Nam, Kyung-Hwa Kim and Henning Schulzrinne
Columbia University, New York, NY
hn2203@columbia.edu, {khkim, hgs}@cs.columbia.edu

***Abstract*—With the proliferation of online video, measuring quality of experience (QoE) has become a pivotal aspect for the analysis of today's over-the-top (OTT) video streaming. To monitor video QoE, we introduce YouSlow that can detect various playback events (e.g., start-up latency, rebufferings and bitrate changes) from video players while a video is being played. Using YouSlow, we have collected more than 400,000 YouTube views from more than 100 countries.**

We measured the impact of these playback events on video abandonment and found that rebufferings incur abandonment rates six times higher than start-up latency, mostly caused by pre-roll ads. A single rebuffering event has three times the impact of a bitrate change. Even increasing the bitrate can raise abandonment rates by a factor of four compared to keeping the bitrate constant.

***Index Terms*—HTTP Video Streaming; Adaptive Bitrate (ABR) Streaming; Video Quality of Experience (QoE)**

I. Introduction

Today's popular video streaming services (e.g., Netflix, Hulu and YouTube) stream video contents to viewers over HTTP or HTTPS. To provide smooth streaming, they use adaptive bitrate (ABR) streaming technologies such as Apple's HTTP Live Streaming (HLS) [1], Microsoft's Smooth Streaming (SS), Adobe's HTTP Dynamic Streaming and Dynamic Adaptive Streaming over HTTP (DASH) [2]. In ABR streaming, a video player dynamically adjusts video bitrates based on estimated network conditions, buffer occupancy and hardware specifications of viewers' devices (e.g., smartphones vs. desktops). Therefore, user-perceived video quality can vary depending on how appropriately the player selects the best available bitrate during a download. As an example, a viewer may experience frequent rebufferings (i.e., a video is paused and then resumes playing repeatedly) when the player requests a higher bitrate than what is actually available in the network. It is also possible for the viewer to be stuck with a low bitrate during the entire playback if the network capacity is underestimated by the player. Hence, from over-the-top (OTT) video service provider's viewpoint, improving ABR heuristics is a key factor to enhancing video QoE.

To improve ABR streaming, it is important to analyze how changing ABR heuristics influences QoE. While traditional quality of service (QoS) based metrics (e.g., measuring TCP throughput, video packet delay and jitter) can be used to pinpoint network impairments, the metrics do not accurately reflect the viewer's watching experience. Thus, we believe that the QoE monitoring system should focus on application-layer events instead of transport-layer events. To achieve this, we suggest monitoring live playback events directly from video players rather than the network middle-boxes such as routers. As a proof of concept, we have developed YouSlow ("YouTube Too Slow!?"), a new QoE monitoring system for OTT streaming services. This lightweight web browser plug-in can monitor various playback events such as start-up latency, rebufferings and bitrate changes directly from ABR players while viewers watch videos on the YouTube web site. So far, YouSlow has collected over 400,000 YouTube views from more than 900 viewers located in more than 100 countries.

In this paper, we evaluate various QoE metrics by analyzing video abandonment rates in YouTube. An abandonment occurs if a viewer closes the video during playback, either due to lack of interest or because they are annoyed by playback events such as long start-up latency, frequent rebufferings and bitrate changes. Our key findings and contributions can be divided into three categories:

- **Development of an analysis tool for video QoE:** YouSlow is designed to detect various playback events while a video is being played. Compared to prior approaches using survey-based metrics, YouSlow saves video researchers time and effort, particularly for large sample sizes. In addition, our QoE monitoring system allows viewers to track their viewing experiences such as statistics of average played bitrates and rebufferings in real time.
- **Development and evaluation of QoE metrics:** We show that tracking rebuffering ratio and bitrate changes during playback is useful to quantify abandonment rates for short videos. We suggest that ABR players should use these metrics to improve user engagement, when switching bitrates and inserting ads in the middle of a playback.
- **An analysis of YouTube:** Our measurements show that rebufferings increase abandonment six times as much as the start-up latency caused (mostly) by pre-roll ads. Most interestingly, our analysis shows that even increasing a bitrate during playback can annoy viewers; when the bitrate changes, they abandon videos more than four times as much. Further, we show that a single rebuffering event can cause abandonment rate three times higher than a single bitrate change.

The remainder of this paper is organized as follows. In Section II, we focus on understanding the principle of ABR

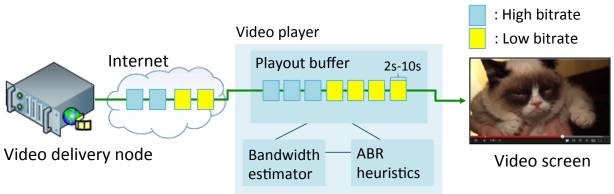


Figure 1: How does ABR streaming work?

streaming, and address the challenges of estimating video QoE using network QoS metrics. Section III describes the overview of YouSlow and its implementation. Then, we present our analysis of YouTube in Section IV. Our QoE analysis report is described in Section V. Finally, we look at the related work and summarize our conclusions in Section VI and VII, respectively.

II. Background

We briefly describe today's HTTP based video streaming technologies. Further, we address the challenges of estimating video QoE using QoS based metrics.

A. Progressive download vs. ABR streaming

Today's video streaming typically uses the two popular HTTP-based streaming technologies: *progressive download* and ABR streaming. Before ABR streaming gained popularity, *progressive download* was used most widely. In *progressive download*, a video server streams a single video file when a video is requested, forcing viewers to watch the same video quality regardless of their local network conditions or playback hardware capabilities. ABR streaming technology was introduced to resolve this problem. In ABR streaming, a video server stores content in multiple resolutions and thus bitrates, and a player selects the best available bitrate regarding various factors such as currently available network bandwidth, remaining playout buffer occupancy, screen resolution and video rendering capability of the viewer's device. Depending on video encoding efficiency and quality, each bitrate is segmented into a series of small constant-duration fragments, varying between two and ten seconds.

As shown in Figure 1, an ABR player consists of three components: a playout buffer, a bandwidth estimator and ABR heuristics. Using a bandwidth estimator, the player monitors available throughput during a download. Based on the download speed and the remaining playout buffer level, ABR heuristics in the player are used to select the best available bitrate while the video is being played. In addition, ABR heuristics take into account hardware specifications of viewers' devices for the bitrate selection algorithm. For example, the player selects high definition (HD) bitrates on desktop browsers while it generally plays standard definition (SD) bitrates on smartphones due to small screen size and low graphics processing unit (GPU) performance.

OTT service providers typically use different fragment duration, playout buffer size and ABR heuristics. Thus, even if the same basic ABR technology is used by several OTT service providers, the performance of ABR streaming can vary.

B. Challenges of analyzing video QoE using QoS metrics

Several researchers [3]–[5] have used QoS-based metrics such as monitoring throughput, goodput, packet delay and jitter from network middle-boxes between viewers' devices and video servers, to estimate video QoE. However, while the metrics can provide approximate QoE, there are still challenges to precisely estimating QoE for buffered video streaming. As an example, low TCP throughput does not always interrupt a viewer's watching experience. Let's suppose that an ABR player has downloaded enough data in the playout buffer. In this case, even if the network has low TCP throughput, the player can still provide smooth streaming until it consumes all data stored in the buffer. Moreover, it is possible for the player to experience low frame rate when a significant number of packets are lost while video rendering, which can degrade QoE of viewers. To avoid this, an ABR player is designed to flush the playout buffer and try to download the entire fragment again. The QoS-based metrics are unable to detect the above events since they cannot accurately track the playout buffer level from the network middle-boxes.

III. Implementation

Unlike prior QoS based metrics, YouSlow can monitor various playback events directly from within an ABR player for an analysis of video QoE. Currently, YouSlow only supports YouTube, but other players' JavaScript APIs such as Vimeo could be easily applied to YouSlow.

Figure 2 shows the architecture of the Chrome plug-in for YouTube analysis. We distribute the extension via the Chrome web store[1] and the YouSlow web site[2]. YouSlow runs in the background of the Chrome browser, and injects our QoE monitoring scripts into the web page whenever a viewer watches a video in the YouTube web site, www.youtube.com[3]. The core scripts contain YouTube player's iframe and JavaScript APIs [6] to access and monitor playback events of HTML5 and Flash video players. When a viewer ends a video session, the extension automatically reports the measurements to our monitoring server[2]. The collected data is analyzed and then marked on Google maps. For privacy reasons, the extension does not collect any information regarding the viewer's YouTube account or video titles.

Through our monitoring system[2], viewers can monitor various metrics about their YouTube watching experiences, such as how often they experience rebufferings and what video bitrates they typically watch. Using this information, they may compare the performance of their own ISPs with other local ISPs. Additionally, YouSlow outputs can be useful to video service providers to improve their ABR streaming services. For example, they can monitor and compare the rebuffering statistics every time there is a change in their ABR heuristics.

[1]Chrome web store - http://goo.gl/AIOED3
[2]YouSlow - https://dyswis.cs.columbia.edu/youslow/
[3]Currently, YouSlow monitors video playback events when viewers watch videos through the YouTube web site only.

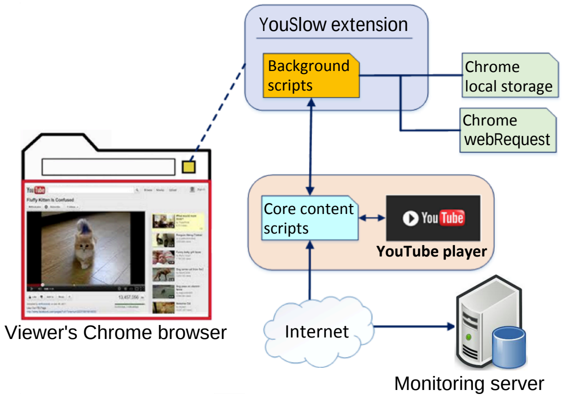


Figure 2: YouSlow Chrome plug-in for YouTube analysis

(a) Top 10 countries

Country [1-5]	Views	Country [6-10]	Views
United States	90,399	Philippines	19,356
India	31,137	Canada	13,722
United Kingdom	29,551	Malaysia	13,225
South Korea	22,648	United Arab Emirates	10,475
Germany	21,358	Brazil	9,629

(b) Top 8 U.S. ISPs

U.S. ISP [1-4]	Views	U.S. ISP [5-8]	Views
Comcast	20,924	Time Warner Cable	6,018
Verizon	11,596	Frontier Communications	5,380
Charter Communications	9,466	Cox Communications	3,342
AT&T	6,203	Qwest Communications	2,096

TABLE I: Dataset

YouSlow can measure the following factors during video playback:

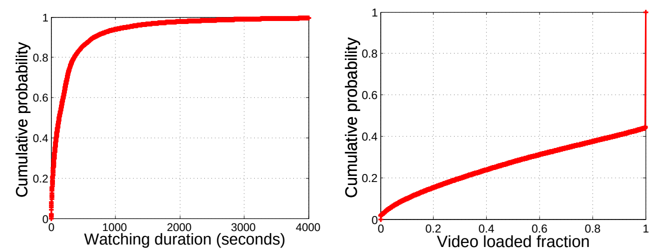
- **Start-up latency:** It measures the start-up delay, which is from the instant a play button is clicked to when the player actually starts to play the main video.
- **Rebuffering:** It monitors the duration of rebuffering and how often it occurs during playback.
- **Bitrate changes:** It measures how much an ABR player increases or decreases the bitrate every time it switches bitrate during playback.
- **Video loaded fraction:** It monitors the percentage of the video that the player shows as buffered. We calculate the fraction by dividing the total amount of downloaded video data by the full size of the video. For example, if the player downloads 10 MB from a 100 MB video, the fraction will be 0.1.
- **Location:** An IP geo-location database[4] is used to pinpoint the approximate location (city, state, and country) of the playback event, and find the domain names of local ISPs that the viewers are connected to.

IV. YouTube Measurements

In this section, we analyze 409,511 YouTube views collected between April 2014 and July 2015, from more than 900 viewers in 102 countries. We note that the dataset only includes the video sessions where the viewers watched videos through YouTube web site using the Chrome browser on desktops or laptops. The Chrome extension does not work on mobile platform such as iOS and Android. Table Ia shows the top ten countries along with the total number of reported views. We also compare and analyze the measurements regarding U.S. ISPs (Table Ib).

Start-up latency: We measure the elapsed time from when a play button is clicked to when the main video starts. There are two factors that contribute to start-up latency: playout buffer and pre-roll advertisements (ads). First, an ABR player has to wait until a certain amount of video data is stored in the playout buffer. Secondly, an ABR player does not play the selected video until viewers watch video ads. According to YouTube's advertising policies [7], there are two types of video ads in YouTube: *skippable* and *non-skippable*. *Skippable* video ads allow viewers to skip the ad after five seconds while *non-skippable* video ads must be watched to view the main video. Both ads can appear before, during or after the main video. We also note that the viewers who use an ad-block extension [8] may be able to watch the entire video without ads. Our tool can distinguish whether the start-up latency is caused by the pre-roll ads or the lack of data stored in the buffer by comparing HTTP URLs using the Chrome webRequest API [9]. We observe that the player uses different URL parameters for downloading the video ads and the main video. Through the measurements, we find that in most cases ($>99\%$) the pre-roll ads are the cause of start-up latency for YouTube videos and its average duration is 6.4 seconds per video session.

[4]Maxmind GeoIP database - http://dev.maxmind.com/

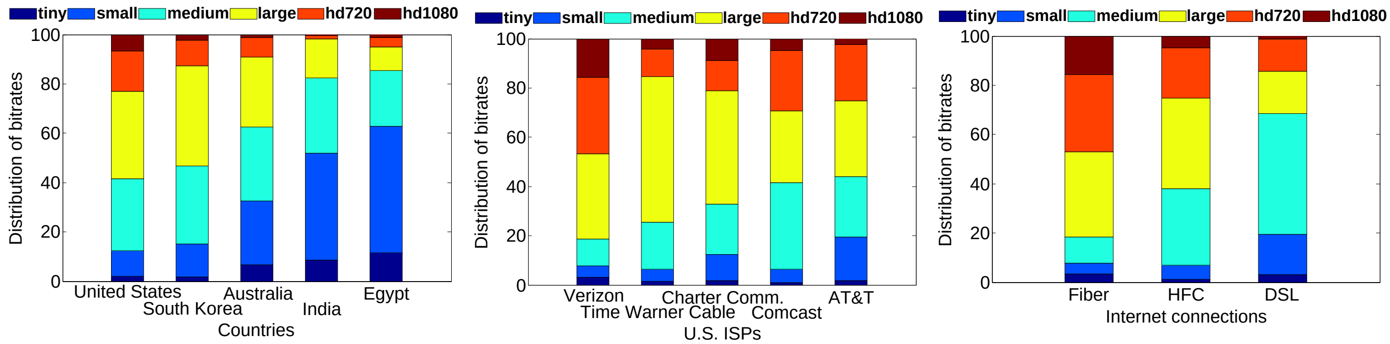


(a) Video watching duration **(b)** Video loaded fraction

Figure 3: Cumulative probability of video watching duration and loaded fraction

Video watching duration: We measure how long a viewer stays in each video session. The watching duration also includes rebuffering and start-up latency. Based on the experimental results in Figure 3a, we observe that the viewers watched YouTube videos for 5:01 minutes on average per video session.

Video loaded fraction: We measure video engagement by monitoring the video loaded fraction described in Section III. According to Figure 3b, more than 40% of viewers closed YouTube videos in the middle of the playback. They may have lost their interest in the videos or suffered from unexpected playback events such as rebufferings and bitrate changes. We will describe QoE assessments in more detail in Section V.



(a) Countries **(b)** U.S. ISPs **(c)** Different types of Internet connections

Figure 4: Comparison of YouTube played bitrates

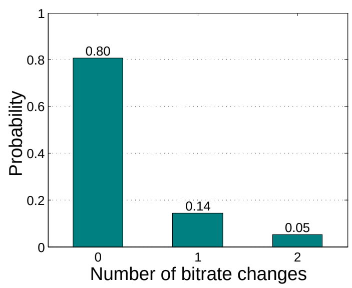


Figure 5: Probability of number of bitrate switches

(a) YouTube bitrate setting

Type	Video bitrate	Resolution
highres	35 - 45 Mbps	3840 × 2160
hd1440	10 Mbps	2560 × 1440
hd1080	8,000 kbps	1920 × 1080
hd720	5,000 kbps	1280 × 720
large	2,500 kbps	854 × 480
medium	1,000 kbps	640 × 360
small	400 kbps	426 × 240
tiny	80 kbps	256 × 144

(b) YouTube played bitrates (%) in dataset

hd1080	hd720	large	medium	small	tiny
3.8%	14%	36.1%	26.8%	16.2%	3.1%

TABLE II: An analysis of YouTube bitrates

Bitrate switches: We observe that most video sessions ($>98\%$) have fewer than three bitrate switches during playback. Figure 5 shows the probability mass function (PMF) of bitrate switches in the dataset. 80% of video sessions in YouTube did not switch bitrates during the entire playback. We observe that the number of bitrate changes and rebufferings are correlated: stable networks do not change bitrates much during a playback, and do not subject the viewer to rebuffering events.

Played bitrates: According to YouTube's encoding policies [10], YouTube streams eight different bitrates: `highres`, `hd1440`, `hd1080`, `hd720`, `large`, `medium`, `small` and `tiny`. We describe each bitrate setting in Table IIa, and measured the distribution of played bitrates in Table IIb. These measurements indicate that most viewers on desktops or laptops watched YouTube videos with `large` (36.1%) or `medium` (26.8%) bitrates. We also observed a few `hd1440` and `highres` videos, but the probability ($<0.1\%$) is much smaller.

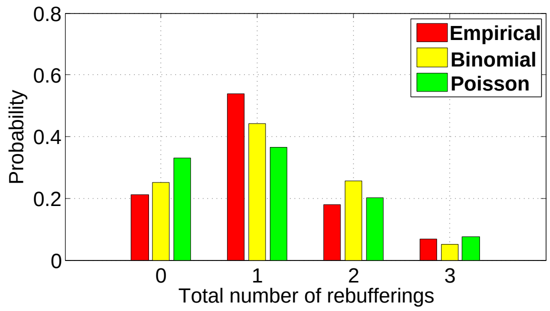
In Figure 4a, we compare the distributions of played bitrates among countries. For example, the viewers in United States and South Korea experienced higher bitrates in comparison to the ones in India and Egypt. Figure 4b shows the distributions of played bitrates for different ISPs in United States. For more details, we compare the distributions depending on different types of Internet connections such as fiber-optic cables, hybrid fiber-coaxial (HFC) and digital subscriber line (DSL). We collected 7,074 samples in total for fiber-optic cables from Verizon's *FiOS* Internet service, and 6,618 samples for HFC from Time Warner Cable, Charter Communications, Cox Communications, Comcast and AT&T's *U-verse* (formerly *Project Lightspeed*). For DSL, we obtained 2,384 samples from Verizon (*non-FiOS*), AT&T (*non-U-verse*) and Qwest Communications. YouSlow can distinguish this by comparing the hostnames of the Internet service providers of the viewers using the IP geo-location database[4]. For example, Verizon uses certain hostnames (e.g., x.x.fios.verizon.net) for the *FiOS* users. Through the measurements, we observe that the viewers using fiber watched more HD bitrates (36.8%) than the ones using HFC (25.3%) or DSL (14.4%).

Rebufferings: In the dataset, we observe that more than 99% of video sessions have fewer than four rebufferings during the entire playback. Figure 6 shows the PMF graphs of total number and total duration of rebufferings per video session. In addition, the figures represent the best fitting distributions. For the number of rebufferings, we find the two distributions which fit the data: *Binomial* (number of trials = 3 and probability = 0.368) and *Poisson* ($\lambda = 1.104$). *Geometric* (probability = 0.243) is the best fitting distribution for the total rebuffering duration. Through these measurements, we find that in most cases, the viewers experienced a small number of rebufferings and the total duration of rebufferings was relatively short.

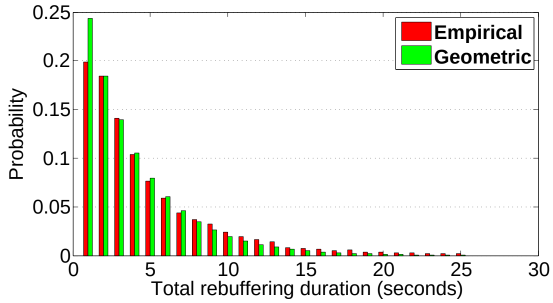
V. Video QoE Analysis

In this section, we describe our analysis of video QoE based on the YouTube measurements in Section IV. We are trying to answer the following questions:

- How do start-up latency, rebufferings and bitrate changes affect viewing interruption?



(a) Probability of total number of rebufferings



(b) Probability of total rebuffering duration

Figure 6: YouTube rebuffering statistics

- What metrics can we use to analyze the impact of the above playback events on video QoE?

A. Analysis methodology and metrics

Unlike Netflix and Hulu that stream long videos such as movies and TV shows, YouTube typically provides short video clips (e.g., music videos and sports highlights). Most viewers access YouTube videos while surfing the Internet and the videos are easily abandoned if the viewers lose interest during playback. We monitor such video abandonment for an analysis of video QoE. Instead of monitoring video packets [11], YouSlow can obtain this data more accurately by monitoring playback status (play, pause and stop) directly from within the video player.

We want to point out how we distinguished the videos that were abandoned due to poor viewing experiences (such as frequent bitrate changes and rebufferings) from the videos that were abandoned due to lack of interest by the viewers or other interruptions unrelated to the viewing experience. The abandonment case lead by rebufferings is quite straightforward. YouSlow can distinguish if a video is paused by the viewer or a rebuffering event. Therefore, when a video is closed while it is paused due to rebufferings, we consider the case as an abandonment. For the case of bitrate changes, it is more complicated. For example, it is difficult to tell if viewers are closing the videos due to bitrate switches or simply because they lost interest, possibly along with the annoyance of bitrate changes. To separate the two causes, we assume abandonment when a viewer closes the video within five seconds of a bitrate change, but only if the viewer has watched at least half of the full content of the video. But since this definition only applies to longer videos, we exclude videos shorter than 30 seconds from our analysis.

Since the video rendering quality and level of interest are independent, we believe that our results are relatively insensitive to changes in defining QoE-driven abandonment. We compute QoE abandonment ratio by dividing the number of sessions abandoned due to impairments by the total number of video sessions. Compared to the metrics based on mean opinion scores (MOS) that can have multiple numbers to reflect service quality (e.g., a numerical value between 1 and 5), YouSlow returns only two values for each video session: 0 (non-abandoned) or 1 (abandoned). There may be exceptional cases where a viewer watches the video to the end even if the viewer had a bad watching experience throughout the entire playback. However, with a large number of samples, we believe that monitoring abandonment rates gives us more practical and reliable outputs to analyze viewing interruptions.

B. QoE analysis report

1) Rebuffering: Most recent studies on video QoE [3], [12]–[15] agree on the fact that rebufferings should be avoided if at all possible to enhance video QoE. In addition, they show that QoE of viewers can vary depending on a rebuffering pattern, i.e., how many or how often rebufferings appear during playback. In this paper, we try to understand how viewers react to such different rebuffering patterns in YouTube, along with abandonment rates. As a baseline analysis, we extract the video sessions from the dataset where the total number of rebufferings is three, and plot the abandonments based on the rebuffering intervals, as shown in Figure 7.

Figure 8 shows our experimental results. Through the measurements, we first find that there are more abandonments when the rebuffering intervals are short. We frequently observe such short rebuffering intervals when an ABR player requests a higher bitrate than what a network can handle. In this case, the video play has to be paused until the player stores a certain amount of data in the buffer, which can cause a series of short-term rebufferings. Furthermore, we observe that an abandonment pattern varies depending on rebuffering intervals. For instance, let's suppose that we have a certain range of first rebuffering interval between 0 and 10 seconds in Figure 8. Depending on the second interval, we clearly see that the distribution of abandonments varies. The question is, how do we normalize the impact of rebuffering intervals and correlate

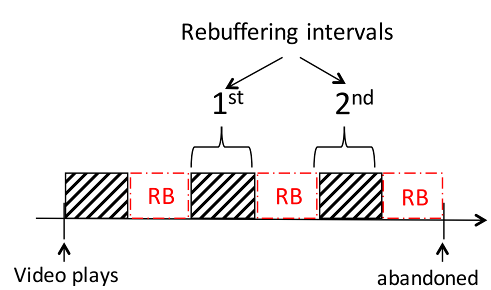


Figure 7: Two rebuffering (RB) intervals with three rebufferings

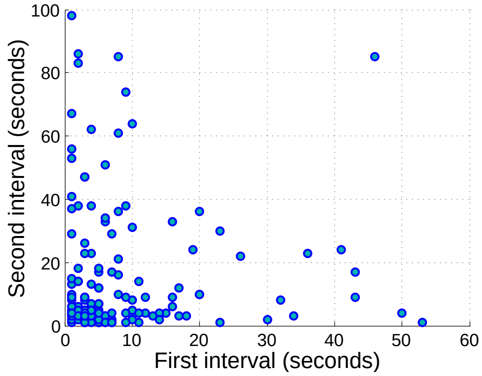


Figure 8: Plotting abandonments regarding two rebuffering intervals

the results with QoE assessments (e.g., MOS)? If we take into account a higher number of rebufferings or additional factors such as rebuffering duration and total playback length, QoE modeling will be much more complicated. To avoid such complexity, we consider a simpler metric below and analyze how the metric predicts the abandonment rate.

Rebuffering ratio: We analyze the impact of rebufferings on abandonment rates using the rebuffering ratio. The ratio is defined as the fraction of time when a viewer experiences rebufferings while watching a video. As an example, let's suppose that rebufferings occur for ten seconds while a viewer watches a 90 second video. In this case, the rebuffering ratio will be $10/(10+90) = 0.1$.

Using the above methodology, we calculate the abandonment rate from the dataset. Figure 9 shows our experimental results. As a baseline analysis, we observe an abandonment rate of 1.2% for video sessions without rebuffering (ratio = 0). We note that the abscissa indicates a range of rebuffering ratio (x - y represents $x < \text{ratio} \leq y$) and the value in parentheses shows the number of samples. For example, we count the number of video sessions with a certain rebuffering ratio (e.g., $0.06 < \text{ratio} \leq 0.08$). There are total of 1,983 samples. Among them, the number of video sessions where the viewers closed the videos during the rebufferings is 83. Therefore, the abandonment rate is 4.2% ($\approx$ 83/1,983). The results tell us that more viewers abandoned the videos as the rebuffering ratio increased.

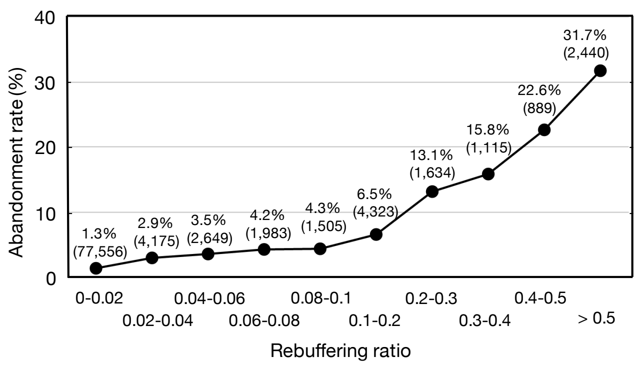


Figure 9: Abandonment rate (%) along with rebuffering ratio

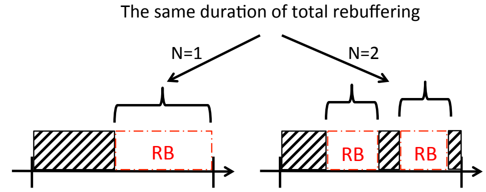


Figure 10: The same total rebuffering (RB) duration with different number of rebufferings

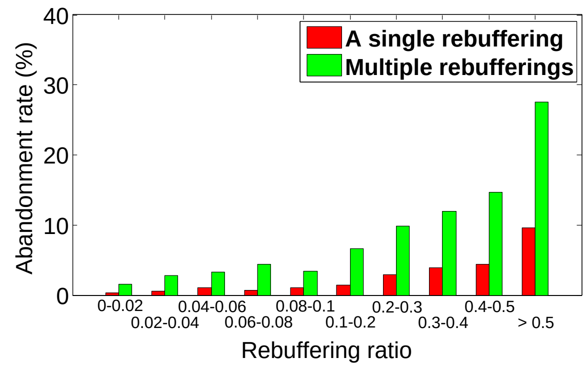


Figure 11: Comparison of abandonment rates between a single rebuffering event and multiple rebufferings

However, we note that the rebuffering ratio does not take the number of rebuffering events into account. As shown in Figure 10, for instance, it is possible that the number of rebufferings can vary although the total rebuffering duration is the same. This can affect video QoE differently. To prove it, we compare the impact of a single rebuffering event and multiple rebufferings by comparing the abandonment rates along with rebuffering ratio. Figure 11 shows our experimental results. We clearly see that an abandonment rate rises in both cases as the rebuffering ratio increases, but multiple rebufferings cause higher abandonment rates compared to a single rebuffering event. Especially when the buffering ratio is larger than 0.5, the abandonment rate for multiple rebufferings is about three times higher than the one for a single rebuffering event.

According to Figure 6a in Section IV, most video sessions in the dataset have a small number of rebufferings (between 1 and 3). We investigate the abandonment rates depending on these numbers. Figure 12 shows our experimental results. Considering the single rebuffering results in Figure 6a, we note that the right side of the graph (e.g., ratio > 0.3) represents the results of the videos with short watching duration and the left side is for the results of videos with a relatively long watching period. For example, the former case is that the viewer closed the video during 10 seconds of rebufferings after watching 10 seconds of the video (ratio = 0.5), and the latter case is that the viewer closed during the same length of rebufferings but after watching 90 seconds (ratio = 0.1). This confirms the results that the abandonment rate varies depending not only on rebuffering duration but also video playback length. We observe that a single rebuffering event shows relatively lower abandonment rate compared to two or three rebufferings, even if they have the same rebuffering ratio.

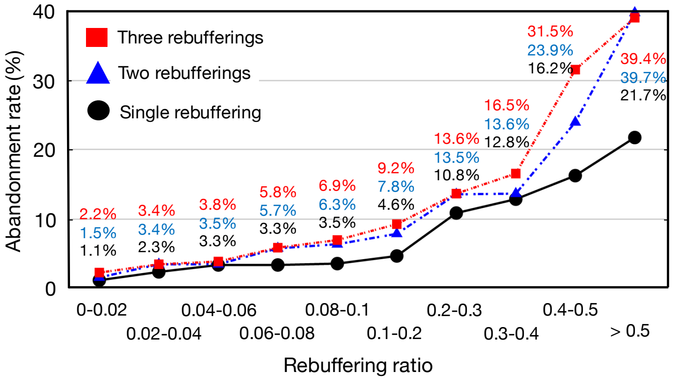


Figure 12: Abandonment rate (%) depending on number of rebufferings between one and three

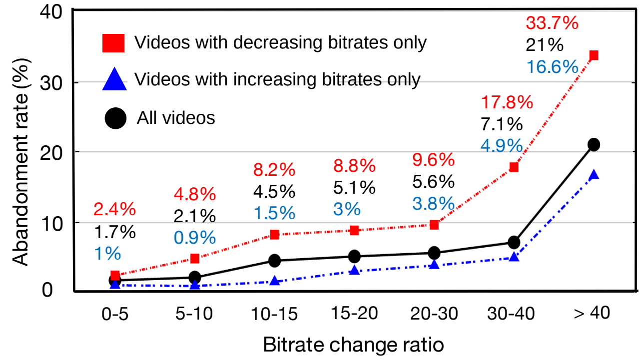


Figure 13: Abandonment rates (%) along with bitrate change ratio

Start-up latency vs. rebuffering: We calculate abandonment rate for different start-up latencies, where the video is closed before, and compare it with the rebuffering case. But the same methodology used for the calculation of rebuffering ratio cannot be used for a start-up latency. The main reason is that during a start-up latency, the ratio is always going to be 1 since the main video never played (e.g., $\frac{latency}{Playback\,time(=0)+latency}$). The ratio will become lower than 1 after the main video starts. In this case, the high ratio means that the video is closed soon after the start-up latency ended. This abandonment typically occurs when the video is not what the viewer intended to watch.

To avoid this problem, we first categorize the dataset into two groups. The first group contains the video sessions where there is only a start-up latency (mostly caused by the pre-roll ads, according to Section IV) and no rebuffering throughout the rest of the playback. In the second group, the video sessions experience a very short start-up latency (<1 second) that viewers are unlikely to notice and only a single rebuffering in the middle of the playback. We take into account only the video sessions with a single rebuffering event so as to avoid the influences caused via multiple rebufferings. In both groups, we count the number of video sessions abandoned by the viewers during either the start-up latency or the rebuffering, and the numbers are divided by the total number of video sessions in each group. As a result, we observe an abandonment rate of 0.6% ($\approx$ 269/44,829) for the first group and 3.9% ($\approx$ 807/20,690) for the second group.

To strengthen the results, it would be better to compare the abandonment rates between the start-up latency caused by initial buffering and the pre-roll ads. Due to lack of samples for the buffering case, however, we leave this for future work. Throughout the above experimental results, we point out that the impact of rebuffering on abandonment rate is more than six times higher than pre-roll ads in YouTube.

2) Bitrate switching: Some papers [16]–[20] investigate the impact of bitrate changes on video QoE. They claim that providing a bitrate as high as possible does not necessarily lead to the highest QoE [17]. They agree on the fact that it is difficult to create a metric that takes into account of all the bitrate switching events, such as the number of bitrate changes, their amplitude (i.e., by how much bitrate increases or decreases) and the duration of each bitrate. Below, we try to find a simple metric that can properly reflect and quantify the impact of bitrate changes on abandonment rates in YouTube.

Bitrate change ratio: To find the impact of bitrate switching on abandonment rates, we take into account absolute values of bitrate changes over playback time using Equation 1.

$$Ratio = \frac{\sum_{i=1}^{Num.\,of\,BR\,changes} |BR_i - BR_{i-1}|}{Total\ playback\ duration\ (second)} \qquad (1)$$

The BR_i and BR_{i-1} denote the newly selected bitrate and the previous bitrate (in kb/s), respectively. Using the above equation, we calculate the abandonment rates. To remove the influence of other factors such as rebufferings and ads, we first collect the video sessions with bitrate changes only. To avoid the case where a video is closed due to lack of interest, we only considered the ones that were watched at least half of the full length and closed within five seconds after the bitrate changes in the middle of a playback. As a baseline analysis, we observe 1.1% of abandonment rate for the video sessions with no bitrate changes (ratio = 0).

Figure 13 shows our experimental results, showing clearly that more viewers abandoned videos as the bitrate change ratio increased. For instance, when the bitrate change ratio is between 30 and 40, the abandonment rate becomes more than four times higher than the case with few bitrate changes ($0 <$ ratio ≤ 5). This result leads to the following question: does switching to a higher bitrate during playback also increase abandonment rate? To figure this out, we analyze the video sessions where the player always switched bitrate to higher ones (e.g., BR_i - $BR_{i-1} > 0$). During the entire playback, in the other words, it never decreased the bitrates. We also calculate the abandonment rates for the video sessions where the player never increased the bitrates during playback. Through the measurements in Figure 13, we clearly see that when decreasing bitrates, more viewers abandoned the videos. Interestingly, we also observe that more viewers abandoned the videos even when the players tried to increase the bitrates. For instance, when the bitrate change ratio is between 30 and 40, the abandonment rate becomes 4.9%, which is more than four times higher than the case with no bitrate change (rate = 1.1%).

3) Rebuffering vs. bitrate switching: We compare the impact of rebuffering and bitrate switching on video abandonment rates. We note that it is difficult to compare both events using the same criteria. For example, we observe that multiple rebufferings can appear as a cluster (Figure 8) while the bitrate seldom changes multiple times in a short period of time. As we described earlier, it would be very complicated to create a proper model to reflect all the factors (e.g., intervals among events and duration for each event) for evaluation. In order to avoid this complexity, we take into account only single events for comparison. We classify the dataset into two groups. In the first group, we collect a total of 9,577 video sessions where the viewers experienced a single rebuffering event without any bitrate changes and any ads. The second group includes a total of 4,991 video sessions that experienced a single bitrate change with no rebufferings and no ads while the video was being played. We use the same methodology to calculate the abandonment rate, and observe 1.2% for videos with a single bitrate change and 3.9% for videos with a rebuffering.

Only a few studies [16], [21] have investigated the comparison between rebuffering and bitrate switching. They conclude that the rebufferings must be absolutely avoided during playback, and that bitrate changes may degrade video QoE when the bitrate switches involving low bitrates. Our solution can quantify the abandonment rates for both events from a large number of samples in YouTube, showing that a single rebuffering event causes abandonment rate three times higher than a single bitrate change.

4) Multiple playback events: So far, we analyzed the impact of rebufferings and bitrate changes on abandonment rates separately. But, both events are not independent, instead correlated in ABR streaming (i.e., a player degrades bitrates to avoid rebufferings). In Figure 14, we combine these two factors together for an analysis of abandonment rate. Overall, the results show that more viewers generally abandon videos as rebuffering and bitrate change ratios increase in the middle of a playback.

We claim that monitoring rebuffering ratio and bitrate changes over playback can be a good reference to improve user engagement while a video is being played. We suggest to implement these metrics in an ABR player, and use the outputs for bitrate selection along with current playout buffer level and available network throughput estimated by the bandwidth estimator (Figure 1). For instance, let's suppose that the current bitrate change ratio is 5 and rebuffering ratio is 0.08. Our goal is to maintain the abandonment rate lower than 10%. In this case, the player may conservatively increase the bitrate (with enough data stored in the buffer and high bandwidth available in the network) unless the bitrate change ratio rises above 10 (Figure 14). It may hold the decision for a certain amount of time if increasing the bitrate pushes the estimated abandonment rate higher than 10%. As another example, the player may track the rebuffering ratio when inserting ads in the middle of a playback. In this case, let's consider the video ad as a single rebuffering event during a download. The idea is to set up the length of mid-roll ads based on the current

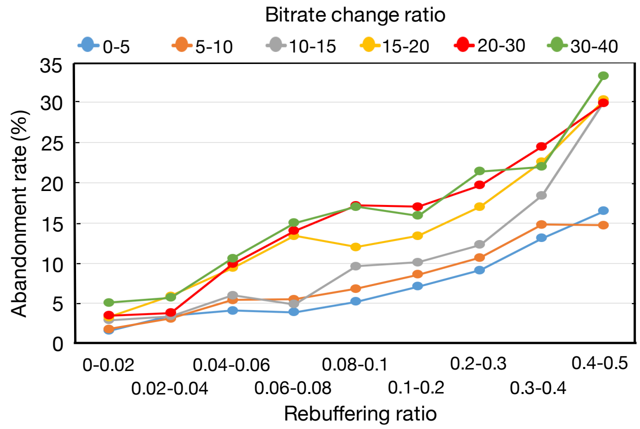


Figure 14: Abandonment rates (%) with multiple playback events

rebuffering ratio. So if the ratio is very low, for example, the viewer is more tolerant of a 30 seconds mid-roll ad. However, it is possible for the longer ads to cause an the rebuffering ratio above 0.2 and the estimated abandonment rate to reach over 10%. In this case, it would be better to give an option of skipping ads after 5 seconds.

C. Summary of Key Observations

These are the key findings from our QoE experimental results:

Finding 1: Our measurements show that a start-up latency in YouTube is mostly caused by pre-roll ads. We conclude that the viewers were more tolerant to such pre-roll ads than rebufferings, with the experimental results showing the impact of start-up latency on abandonment rates to be six times lower than the impact of rebufferings in YouTube.

Finding 2: We observe that viewers are more likely to abandon videos with multiple rebufferings compared to a single rebuffering, event although the rebuffering ratio is the same. We also confirm that viewers prefer constant bitrate to increasing bitrate during playback, and frequent bitrate increase can lead to an abandonment rate more than four times higher than a case with no bitrate changes.

Finding 3: According to our analysis, a single rebuffering event causes abandonment rate three times higher than a single bitrate change.

Finding 4: We show that monitoring rebuffering and bitrate change ratios is a good metric to quantify video abandonment rates for short videos such as YouTube. We suggest to implement these metrics in an ABR player to improve user engagement especially when inserting video ads or changing bitrates in the middle of a playback.

VI. Related Work

Google video quality report [22] provides statistics of YouTube played bitrates along with local ISPs. The methodology is to calculate goodput and rate the ISP performance by comparing the measurement with pre-defined thresholds. However, the output does not provide any QoE factors from the perspective of viewers, such as how often they experience bitrate changes and rebufferings. Dobrian et al. [23] at Conviva monitored user-engagement based on various playback events measured from video players. The methodology used for data collection is similar to our approach. But our platform allows viewers and video service providers to monitor various playback statistics in real time via our QoE monitoring system. In addition, we suggest simpler metrics (e.g., monitoring rebuffering ratio and bitrate change ratio over playback time) that can be implemented at video players to estimate abandonment rates. We believe that the measurement can be a good reference to improve ABR streaming, when changing bitrates or inserting ads in the middle of a playback.

For analyzing network performance issues such as page loading times, Dhawan et al. [24] introduce *Fathom*, a browser-based network measurement platform. As a proof of concept, they have built a Firefox extension that allows web sites or other parties to program network measurements using JavaScript. They introduce case studies of using the platform, but do not investigate QoS or QoE metrics. Shafiq et al. [11] monitored video abandonment by inspecting video packets from ISPs' viewpoint, but the method is not simple compared to our web browser plug-in that can detect such abandonments directly from video players. Hossefeld et al. [13] investigated the impact of rebuffering patterns (i.e., how many and often rebufferings appear during playback) on video QoE. Ni et al. [19] study how viewers experience bitrate changes at different amplitudes and frequencies. Using HTTP DASH, Mok et al. [17] show that viewers prefer to change bitrate gradually. YouSlow provides a cost-effective way to collect a large number of samples and confirms various QoE metrics with evidence from large video streaming services such as YouTube.

VII. Concluding Remarks

We introduced YouSlow as a new video QoE analysis tool for video QoE. This lightweight web browser plug-in can detect various playback events for an analysis of video QoE. Our experimental results show that monitoring rebuffering ratio and bitrate changes over playback time is a proper QoE metric to analyze abandonment rates for short videos such as YouTube. As key observations, we find that a start-up latency mostly caused by pre-roll ads have less impact on abandonment rates, compared to rebufferings. Further, our analysis shows that viewers prefer constant bitrate to increasing bitrate during playback, and a single rebuffering causes abandonment rate three times higher compared to a single bitrate change. We believe that our proposed QoE metrics and experimental results give us an insight to improving ABR heuristics embedded in ABR players and enhancing viewing experiences.

VIII. Acknowledgments

We would like to thank Jung Woo Lee and Jongmee Kim for their valuable feedback and assistance. Kyung-Hwa Kim was supported by the National Science Foundation under Grant No. CNS-1218977.

References

[1] Roger Pantos and Jr. William May. HTTP Live Streaming. Internet Engineering Task Force (IETF) Internet-Draft, April 2011.

[2] Stefan Lederer, Christopher Müller, and Christian Timmerer. Dynamic Adaptive Streaming over HTTP Dataset. In *Proceedings of ACM MMSys*, Chapel Hill, North Carolina, USA, February 2012.

[3] R.K.P. Mok, E.W.W. Chan, and R.K.C. Chang. Measuring the Quality of Experience of HTTP Video Streaming. In *Proceedings of IFIP/IEEE IM*, Dublin, Ireland, May 2011.

[4] Kuan-Ta Chen, Chi-Jui Chang, Chen-Chi Wu, Yu-Chun Chang, and Chin-Laung Lei. Quadrant of Euphoria: A Crowdsourcing Platform for QoE Assessment. *IEEE Network*, 24(2):28–35, March 2010.

[5] P. Casas, R. Schatz, and T. Hossfeld. Monitoring YouTube QoE: Is Your Mobile Network Delivering the Right Experience to your Customers? In *Proceedings of IEEE WCNC*, Sanghai, China, April 2013.

[6] YouTube APIs. Retrieved Apr. 21, 2015 from http://goo.gl/8WMsxS.

[7] YouTube video ads. Retrieved Apr. 14, 2015 from http://goo.gl/CnyLV8.

[8] Chrome and Safari AdBlock. Retrieved Apr. 16, 2015 from https://getadblock.com/.

[9] Chrome extensions - webRequest API. Retrieved Apr. 20, 2015 from https://developer.chrome.com/extensions/webRequest.

[10] YouTube live encoder settings, bitrates and resolutions. Retrieved Apr. 14, 2015 from http://goo.gl/7Jst2B.

[11] Muhammad Zubair Shafiq, Jeffrey Erman, Lusheng Ji, Alex X. Liu, Jeffrey Pang, and Jia Wang. Understanding the Impact of Network Dynamics on Mobile Video User Engagement. In *Proceedings of ACM SIGMETRICS*, Austin, Texas, USA, June 2014.

[12] A. Floris, L. Atzori, G. Ginesu, and D.D. Giusto. QoE Assessment of Multimedia Video Consumption on Tablet Devices. In *Proceedings of IEEE Globecom QoEMC Workshop*, Anaheim, California, December 2012.

[13] Tobias Hossfeld, Dominik Strohmeier, Alexander Raake, and Raimund Schatz. Pippi Longstocking Calculus for Temporal Stimuli Pattern on YouTube QoE: 1+1=3 and $1{\cdot}4{\neq}4{\cdot}1$. In *Proceedings of ACM MMSys MoVid Workshop*, Oslo, Norway, February 2013.

[14] T. Hossfeld, S. Egger, R. Schatz, M. Fiedler, K. Masuch, and C. Lorentzen. Initial Delay vs. Interruptions: Between the Devil and the Deep Blue Sea. In *Proceedings of IEEE QoMEX*, Melbourne, Australia, July 2012.

[15] T. De Pessemier, K. De Moor, W. Joseph, L. De Marez, and L. Martens. Quantifying the Influence of Rebuffering Interruptions on the User's Quality of Experience During Mobile Video Watching. *IEEE Transactions on Broadcasting*, 59(1):47–61, March 2013.

[16] Liu Yitong, Shen Yun, Mao Yinian, Liu Jing, Lin Qi, and Yang Dacheng. A Study on Quality of Experience for Adaptive Streaming Service. In *Proceedings of IEEE ICC 2013*, Budapest, Hungary, June 2013.

[17] Ricky K. P. Mok, Xiapu Luo, Edmond W. W. Chan, and Rocky K. C. Chang. QDASH: A QoE-aware DASH System. In *Proceedings of ACM MMSys*, Chapel Hill, North Carolina, February 2012.

[18] A. Rehman and Zhou Wang. Perceptual Experience of Time-varying Video Quality. In *Proceedings of IEEE QoMEX*, Klagenfurt, Austria, July 2013.

[19] Pengpeng Ni, Ragnhild Eg, Alexander Eichhorn, Carsten Griwodz, and Pål Halvorsen. Flicker Effects in Adaptive Video Streaming to Handheld Devices. In *Proceedings of ACM Multimedia*, Scottsdale, Arizona, USA, November 2011.

[20] David C. Robinson, Yves Jutras, and Viorel Craciun. Subjective Video Quality Assessment of HTTP Adaptive Streaming Technologies. *Bell Labs Technical Journal*, 16(4):5–23, 2012.

[21] A.K. Moorthy, Lark Kwon Choi, A.C. Bovik, and G. de Veciana. Video Quality Assessment on Mobile Devices: Subjective, Behavioral and Objective Studies. *IEEE Journal of selected Topics in Signal Processing*, 6(6):652–671, October 2012.

[22] Google Video Quality Report. Retrieved Jul. 24, 2015 from http://www.google.com/get/videoqualityreport/.

[23] Florin Dobrian, Vyas Sekar, Asad Awan, Ion Stoica, Dilip Joseph, Aditya Ganjam, Jibin Zhan, and Hui Zhang. Understanding the Impact of Video Quality on User Engagement. In *Proceedings of ACM SIGCOMM*, Toronto, Ontario, Canada, August 2011.

[24] Mohan Dhawan, Justin Samuel, Renata Teixeira, Christian Kreibich, Mark Allman, Nicholas Weaver, and Vern Paxson. Fathom: A Browser-based Network Measurement Platform. In *Proceedings of ACM IMC*, Boston, Massachusetts, USA, November 2012.