Impact of Duration on Active Video testing

Saba Ahsan
Aalto University
saba.ahsan@aalto.fi

Varun Singh
callstats.io
varun@callstats.io

Jörg Ott
Technische Universität München
ott@in.tum.de

ABSTRACT

There is a growing interest in video performance measurements with emphasis on user experience and several initiatives have been taken to conduct active testing of real video services. A deeper understanding of the variations in media bit rate and its influence on the performance of video playback is needed in order to design better measurements. In this paper, we analyze a dataset of YouTube videos from various genres. We show statistically that most YouTube videos can be represented sufficiently well by the first 1 to 3 minutes of the video. This eliminates the need for running longer tests when network conditions are stable as in the case of fixed networks. We test our observation in an active testing environment that measures video metrics, and recommend based on the results that such tests should run at least for one minute, however, a duration of 3 minutes will help achieve better and more stable results.

CCS Concepts

•Information systems → Multimedia streaming; •Networks → Network measurement;

Keywords

HTTP streaming; Large-scale measurements; Active measurements

1. INTRODUCTION

Video has been the dominant traffic on the Internet for some time and is expected to rise even further in coming years. It has been estimated that IP video traffic will increase to 80% by 2019 in comparison to the 67% in 2014 [8]. Many users, collectively referred to as "cord cutters", are actively switching to Internet video streaming as their primary form of entertainment. Furthermore, high bit rate content such as Ultra HD (4K) greatly increases the impact and requirements of media traffic on the Internet. Given such trends, there is a need for improved methods of delivery for video, and additionally for performance testing and monitoring to ensure a good user experience. This would require designing measurements that focus specifically on video user experience.

NOSSDAV'16, May 13 2016, Klagenfurt, Austria

DOI: http://dx.doi.org/10.1145/2910642.2910651

ISPs and application service providers have a strong interest in assessing and understanding network and application performance to make sure that their customers are satisfied. Hence, performing active measurements at the endpoints is becoming an important tool for observing long-term network behavior, as well as for investigating and diagnosing network failures. Measurement endpoints include infrastructure nodes such as access routers and set-top boxes as well as user devices such as, personal computers, smartphones, and tablets. Typical metrics, e.g., as defined by the IP Performance Metrics (IPPM) Working Group[1] are round trip delay, one way delay, IP packet delay variation, average TCP/UDP throughput, average fractional loss, DNS latency, among others. Aggregating performance metrics from many measurement points by an Internet Service Provider (ISP), or a measurement service (e.g., RIPE Atlas[2], SamKnows[3], Netradar[4] [22], Speedtest[5], etc.) allows characterizing the network performance geo-spatially and over time, diagnose outages and observe the impact of the outage, and lastly, the collected information helps regulators develop better public policy for the Internet.

In this paper, we explore the characteristics of Internet video to facilitate the design and execution of active measurements. Such active measurements are used for measuring performance at the end point, suitable for large scale measurements (as defined by the IETF LMAP[6] Working Group). They are also applicable for testing new video applications or adaptation algorithms for HTTP Adaptive Streaming (HAS). Given the variety in types of content, encoding algorithms and parameters, video streams have high variability. This variability can have an impact on video performance especially when the network resources are limited. Recent popularity of HTTP Adaptive Streaming(HAS), have further increased demands by requiring each video to be encoded in a number of bit rates resulting in multiple representations for a single video. It is important to clarify here that the term bit rate in this paper is used in reference to the video bit rate and not throughput levels observed on the network. The purpose of this study is to find a balance between the time limitations of active testing and more meaningful video performance statistics. To this end, we analyze a dataset of popular YouTube videos based on charts from 58 different locations and make the following contributions:

1. We show that the magnitude and variation of bit rates for YouTube videos can be represented by a short clip from the

[1]https://datatracker.ietf.org/wg/ippm/
[2]https://atlas.ripe.net/
[3]http://www.samknows.com/broadband/
[4]https://www.netradar.org/
[5]http://www.speedtest.com/
[6]https://datatracker.ietf.org/wg/lmap/

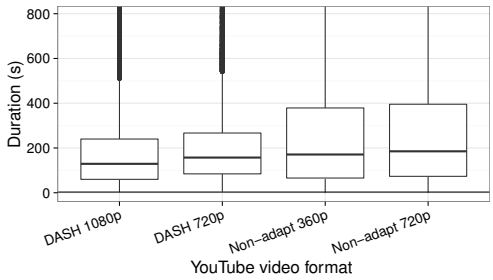

Figure 1: The distribution of video duration in seconds of the videos collected for this study. The extremes of the upper whiskers are not shown; DASH datasets include videos over an hour long while non-adaptive videos include length of 11 hours.

beginning of the video, typically 1 to 3 minutes long.

2. We provide results from experiments that monitor user experience metrics during active video testing over HTTP, which show that cutting off tests prematurely produce reliable results when the cut-off duration is at least 1 minute. Results are further improved, however, for a cut-off of 3 minutes.

We chose YouTube for our analysis for two reasons: ease of access without logging in and its popularity. According to Sandvine Global Internet Phenomena Report, YouTube was the largest single source of real-time entertainment traffic for both mobile and fixed access networks and the largest contributor of Internet traffic in the world in 2013 [21]. SamKnows *Whitebox* that measures broadband performance through active testing includes a YouTube test in the testing suite. The white boxes are being used in different countries for large scale active measurements on behalf of regulators. The same test is used in our active testing experiments with slight modifications to make it suitable for our experimental setup.

The rest of this paper is organized as follows. We describe related work in section 2. Section 3 describes our datasets and the methodology used for data collection. Statistical analysis of our datasets for selecting a suitable length of clip that can represent the video is given in section 4. A validation of our finding from the analysis using an experimental setup for video testing is described in section 5. Finally, we present our conclusions in section 6.

2. RELATED WORK

Several studies have been conducted to characterize YouTube videos using datasets collected in 2007-08 [1,7,13] and in 2013 [3]. Initially in 2007, YouTube had a size limit on videos of 100MB [13], which has since been increased to 20GB [23]. Our datasets were collected between 2013 to 2015 and contain Full HD (1080p) and Dynamic Adaptive Streaming over HTTP (DASH) streams.

In [5], the authors use over 20 million randomly selected YouTube videos to show that the popularity of videos is constrained by geographical locations. Our methodology is in line with this observation, and our dataset contains all available location-based charts from YouTube, giving our dataset a regional representation.

A crowdsourcing study in [14] shows that the QoE for TCP video streaming is directly related to the number and duration of stalls during a video playout, while other factors like age, level of Internet usage or content type have no significant impact. In [6], the authors build a QoE model based on stalling events for YouTube. Research has also shown that actively measuring stall events (with the Pytomo tool [15]) in different Internet Service Providers (ISP) helps predicting the user experience [19]. The authors of [18] have developed a web-browser plugin that reports YouTube performance based on occurence of stall events.

Another study analyzed a dataset from of an adult video streaming website and it was observed that popularity of content had a dependency on the content metadata and the ease of access of the video [24]. In [4], the researchers study how YouTube's progressive download leads to TCP packet losses. The impact of location, devices and access technologies on user behavior and experience is discussed in [12]. Distribution of YouTube's cache servers and their selection process was studied in [2]. A comparison of YouTube datasets collected using different sampling methods showed that some sampling methods introduce bias and subsequently affect the results [16].

Our work aims at active measurements and is thus relevant to the LMAP [11] and IPPM WGs.

ITAG	Resolution	Format	Video Codec	Audio Codec	Sample
22	720p	MP4	H264	AAC	32133
18	360p	MP4	H264	AAC	62387
137	1080p	MP4 DASH	H264	-	8167
136	720p	MP4 DASH	H264	-	19618
135	480p	MP4 DASH	H264	-	13(films)
134	360p	MP4 DASH	H264	-	15(films)

Table 1: ITAG values of YouTube videos discussed in this paper and the size of the sample (number of videos) for each type used in the analysis presented in Section 4. Itags 134 and 135 were used only for feature-length film analysis.

3. DATASET

We developed a YouTube client designed to mimic playout of YouTube videos and read video containers. The client downloads videos from YouTube and collects frame level information from the streams: the timestamp at which the frame is to be played and its size in bytes. YouTube uses numeric identifiers called itags for identifying the formats and resolutions of the video, and the itags used by this study are listed in Table 1. DASH videos are available in different representations, each with its own encoding bit rate and resolution to provide a range of video qualities to the client. The resolution in the table indicates the vertical resolution and the "p" is for progressive scan. The horizontal resolution will depend on the aspect ratio of the video and YouTube players add black bars on the sides if needed to display them correctly. Each representation is divided into chunks of a fixed duration, which in the case of YouTube is 5 seconds. A DASH client can typically adapt to network conditions by switching between representations at the boundaries of these chunks to ensure smooth playback. Hence, the final played stream depends on the network conditions and the adaptation algorithm. Since we wanted to study the streams without network and algorithm dependencies, we did not use any rate-adaptation algorithm in the client when downloading DASH and instead downloaded entire streams in a single resolution.

We collected *frame-logs*, constituting frame sizes and timestamps, for YouTube MP4 files in four different resolutions and for both DASH and non-adaptive streams. The resolutions gathered for DASH were 360p, 480p, 720p, and 1080p, while those for non-adaptive were 360p and 720p. YouTube did not offer non-adaptive streams for 480p and 1080p resolutions in MP4 at the time that this study was conducted. All videos used were from the charts of July 5 and September 11, 2013 [7]. These charts are auto-generated by

[7]www.youtube.com/charts.YouTube has changed their service since the collection of the data and now redirects chart requests to "Popular on YouTube" channel.

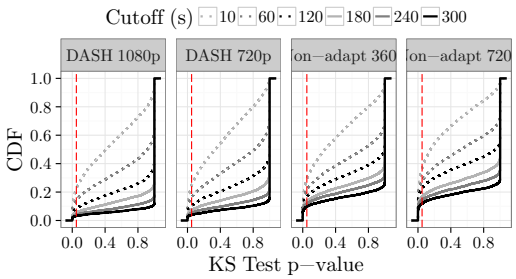

Figure 2: The distribution of p-values of KS test for different cut-off lengths and formats. For a 3 minute clip (cut-off = 180s) over 90% DASH videos and at least 80% of Non-Adaptive videos pass the test.

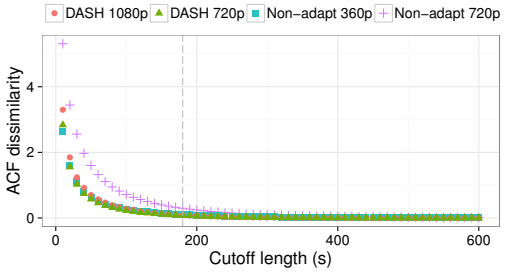

Figure 3: The dissimilarity based on ACF for the segment and the full video for different formats. The points on the graph represent the 90th percentile of the dissimilarity value for a particular cut-off length. The x-intercept is drawn at 180 seconds.

YouTube for a set of locations based on daily, weekly, monthly or all time number of views. Apart from the overall top videos, we collected data from category based charts for all available locations to give us a wide range of video types. Note that the *frame-logs* were collected only for a subset of the videos from the charts, however, the videos were not selected based on any specific content characteristics, but instead the reduced number was due to availability in required formats or failure to retrieve video due to e.g. restricted access to the video at the time of the test. Details of the dataset can be found in [3].

4. VIDEO TESTING DURATION

Video streams compressed using Variable Bit Rate (VBR) encoding schemes are bursty by nature with high motion scenes producing higher bit rate values in comparison to low motion scenes. The extent of burstiness depends on the content of the video and the encoding. This characteristic of video streams can affect the performance of video especially when the network requirements of the video are close to the available network resources. This makes it beneficial to run video tests for longer periods. On the other hand, active video measurements over live networks need to be kept short to minimise interference with actual traffic. Furthermore, creating and maintaining test streams of long durations is difficult. Performance measurements for video, therefore, call for a balance between tests that are neither too long, nor too short. In this section, we look at the characteristics of video streams from YouTube to determine whether or not cutting off a video before it ends yields a video clip that represents the characteristics of the entire video sufficiently well. We use two statistical measures to guide our choice of a cut-off value for the clip: 1) the two sample Kolmogorov-Smirnov (KS) test acceptance rate 2) the Autocorrelation Function (ACF)-based dissimilarity values. The clip is always taken from the beginning of the video and consists of all frames with timestamp t < cut-off value. When the selected cut-off value is longer than the duration of the video, the clip will be identical to the video. The size and type of datasets used for the analysis are shown in Table 1 and the distribution of the video lengths of different formats within the dataset is shown in Figure 1. The statistical analysis is done using R software environment and libraries [20] [17].

4.1 KS Test

The two-sample KS test [9] is a goodness of fit test that takes two mutually independent, random samples, X and Y with values $X_1, X_2, X_3, ..., X_n$ and $Y_1, Y_2, Y_3, ..., Y_m$ respectively and unknown distribution functions. The null hypothesis is that both samples have the same distribution function and it is tested based on the distance between the Empircal Cumulative Distribution Function (CDF) of X and Y.

In our analysis, the KS test is used for comparing a video with a clip from the same video; X is the instantaneous bit rates of the entire video and Y is the instantaneous bit rates of the clip. The instantaneous bit rates are calculated per second using the frame sizes in the stream. For instance, the first value will be the sum of the frame sizes in kbits of all the frames with a playout timestamp less than 1 second (s), the second value will be the sum of the frame sizes in kbits of all the frames with a playout timestamp greater than or equal to 1s but less than 2s and so on.

We did the analysis for 4 types of videos: 360p non-adaptive MP4 (itag 18), 720p non-adaptive MP4 (itag 22), 720p DASH MP4 (itag 136) and 1080p DASH MP4 (itag 137). We also use different cut-off lengths ranging for 10 seconds to 10 minutes. The output of the KS-test is a p-value, which is the probability that the null hypothesis is true. A higher p-value is desirable in this case. Figure 2 shows the distribution of p-values for some of the cut-off lengths, subset by the kind of video. Note that the CDF shown in the graph is a depiction of the distribution of the p-values for the entire dataset and not for any individual video. If we use a significance level of 0.05, accepting the null hypothesis if p-value>0.05, we see an acceptance rate close to 90% for both DASH formats at a cut-off of 180s. In the figure, the intersection of the CDFs with the dashed 0.05 intercept marks the percentage of videos for which the hypothesis fails. The value is lower for the non-adaptive rates, however, the difference is most likely because of the difference in the lengths of the videos in our datasets for each video type rather than the encoding. Figure 1 shows the distribution of video lengths for each format, and the non-adaptive itags have a wider range of video lengths.

We observed that for lower cut-off, there is a larger impact on the acceptance rate of the hypothesis than for higher cut-off values. So in Figure 2, the CDFs of cut-off lengths 10s, 60s and 120s are more widely spaced whereas the cut-offs for 180s, 240s and 300s are closely spaced. We infer from this that after a certain length, increasing the cut-off value further does not add a lot of information in terms of the required bit rate values and variations of a video stream.

The KS test has an obvious disadvantage in the context of video streams because it treats the frame sizes as an unordered distribution and, thus, fails to account for the timing element. Although, using bit rate values instead of raw frame sizes preserves some level of order, to reinforce our finding we look at dissimilarity based on ACF of the video streams to account for the temporal structure of the stream.

4.2 ACF based dissimilarity Test

To measure the dissimilarity between the clip and the entire video while maintaining the order of the time series, we use a dissimilarity measure based on its ACF. This method allows comparison of two video streams of different lengths. P.D'Urso et. al. define autocorrelation as *"a measure of how well a signal matches a time shifted version of itself, as a function of the amount of time shift which is also referred to as a lag"* [10]. Formally, if $x = x_t : t = 1, ..., T$ is a time series then the autocorrelation at lag $r (r = 1, ..., T - 1 = R)$ is defined as

$$\hat{\rho_r} = \frac{\sum_{t=r+1}^{T}(x_t - \bar{x})(x_{t-r} - \bar{x})}{\sum_{t=1}^{T}(x_t - \bar{x})}$$

The ACF of a time series is the series of autocorrelation at different lags. To formally define the ACF-based dissimilarity, suppose two time series are X_T and Y_T respectively and $\hat{\rho}_{i,X_T}$ $\hat{\rho}_{i,Y_T}$ is the estimated autocorrelation value at lag i for X_T and Y_T respectively, then the dissimilarity d_{ACFU} is defined as follows [17]

$$d_{ACFU}(X_T, Y_T) = \sqrt{\sum_{i=1}^{L}(\hat{\rho}_{i,X_T} - \hat{\rho}_{i,Y_T})^2}$$

where L is the maximum lag in the ACF.

In our analysis, we use the clip and the entire video as the two time series; both consist of an ordered series of frame sizes in bytes. The maximum lag used is 100. Figure 3 shows the computed ACF-based dissimilarity of the cut-off clip and the original video for different cut-off values. We use the 90^{th} percentile of the observed values to eliminate the effect of the large amount of videos shorter than the cut-off lengths, which will have 0 dissimilarity and will heavily reduce the average dissimilarity (see Figure 1). While the actual dissimilarity values depend on the chosen lag and is insufficient to draw a conclusion from, the shape of the graph shows a clear decreasing behavior that levels out for higher cut-off lengths and further increasing the cut-off length has a minimal impact on the dissimilarity. We experimented with different lag values, and observed similar shaped graphs.

4.3 Feature-length films

YouTube primarily serves short videos and our datasets for all formats have a large number of videos that are no longer than 3 minutes. Furthermore, many longer YouTube videos have repititive content or at least similar content. Feature films, on the other hand, run for over an hour and often have different types of content, low motion and high motion scenes, throughout the film. In order to compare the validity of the results with feature-length films, we used YouTube's *Films* channel and studied DASH, MP4 bit rates for 15 videos from the free films category [8] that had unrestricted access. The complete list of studied films along with title, genre and durations is available online[9]. Due to limited availability of higher resolution videos, we only performed the tests for itag 134 and 135, which are DASH 320p and 480p respectively. Figure 4 shows the results of the ACF-based dissimilarity computations for different cut-off lengths. The graph has a similar shape to what we observed before, showing that after a certain cut-off, further increasing the cut-off length decreases the dissimilarity measure by a small amount. However, the KS test results shown in Figure 5 show that the p-value is sometimes higher for shorter cut-offs than longer ones. We expect such results for videos that have different types of content, since KS test considers CDFs, which do not take ordering into account. A clip of a low motion scene might yield far too many low bit rate values, while that of a high motion scene would yield too many high bit rate values in comparison to the distribution of bit rates within the entire video.

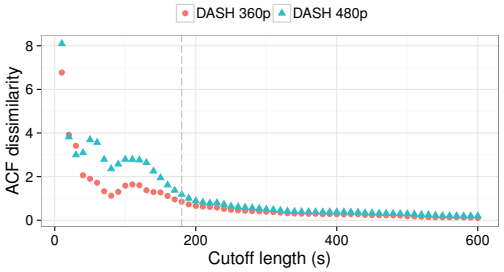

Figure 4: The dissimilarity based on ACF for the segment and the full video for different formats of feature length movies. The points on the graph represent the 90th percentile of the dissimilarity value for a particular cut-off length. The x-intercept is drawn at 180 seconds.

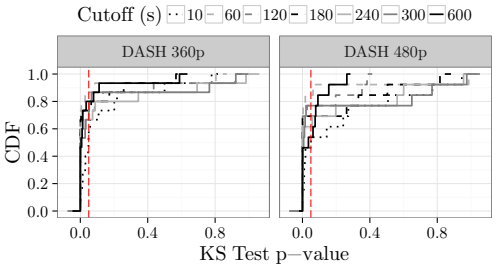

Figure 5: The p-values for KS test of feature length films are less stable. The behavior is most likely because the type of content in films, unlike most YouTube user-generated videos, is not consistent throughout the stream. Since our dataset has less than 15 videos in each format, it is not possible to draw a behavior pattern, however, it is sufficient to see that the clips are not suitable in many cases.

5. VALIDATION

Our statistical analysis shows that a clip of 1 to 3 minutes can be used while conducting YouTube video testing, because the clip can represent the bit rate variations within the entire stream relatively well. We validate this result using a series of tests in a test environment with varying network statistics, showing that after 1 minute, the video metrics begin to stabilize.

5.1 Test Setup

Our test setup consists of an HTTP server that serves video content and a video client that streams it. We connected the two over LAN and emulated link speeds using Netem[10]. The queue size limit was set to produce a maximum latency of 70ms for all cases. All tests are terminated after 30 minutes.

The server runs an HTTP server based on libmicrohttpd [11]. It handles the *range* keyword within the requested URL to serve byte ranges (chunks) within the video files to facilitate progressive download or DASH chunk downloads. The video client is a C/C++ pro-

[8] https://www.youtube.com/user/movies/videos?view=26
[9] http://www.netlab.tkk.fi/tutkimus/rtc/ListofTestVideos
[10] http://www.linuxfoundation.org/workgroups/networking/netem
[11] http://www.gnu.org/software/libmicrohttpd/

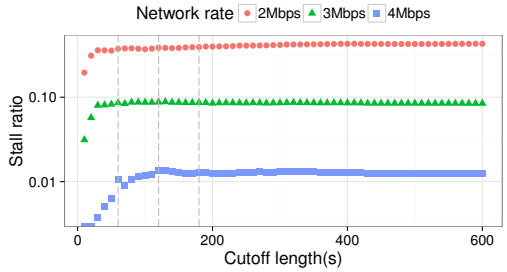

Figure 6: The average stall ratio (number of stalls/duration of video) for YouTube's popular videos of different lengths (range: 15s -95m) showing that results stabilize after 60s. The y axis uses a log10 scale to magnify the axis for the 4Mbps network.

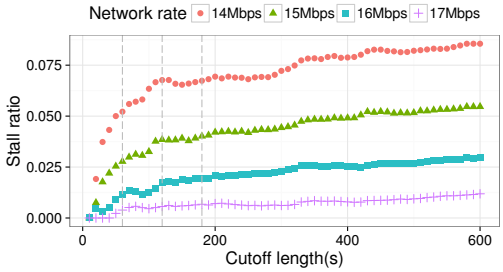

Figure 7: The average stall ratio (number of stalls/duration of video) for YouTube's popular videos where duration is over 10 minutes. Note that for 17Mbps network, the stall ratio is 0 for cut-off under 50 seconds.

gram based on libcurl and is designed to read frame boundaries and playout timestamps to determine the quality of video playback.

The measurements are all done within our video client. The client performs 3 basic functions to mimic a real video player 1) prebuffering, 2) throughput throttling 3) rebuffering. The client operates in prebuffering mode until it downloads at least 2 seconds of video. It then starts the playout timer, and continues to download videos until its buffer is full. We use a buffer of 50 seconds during our trials, which is in compliance with what Safari and Chrome players used for YouTube videos at the time of writing this paper. While the buffer is still filling up, the client requests the video in chunks of 2.5MB. Once the buffer is full, the client slows down the rate of download by requesting 500kB chunks only when the buffer has space. This behavior results in throughput throttling and is used in clients to avoid unnecessary downloading. The requested chunk sizes are calculated assuming an average bit rate of 4Mbps of the video, hence downloading 5 second chunks while the buffer is empty and then throttling down to 1 second chunks. The third function, rebuffering, comes into play when the buffer is empty during playout causing the video playout to stall. If this happens, the client stops the playout timer and resumes only once it has downloaded/rebuffered 1 second of video. Actual rendering of the video is not done.

5.2 Metrics

To measure video experience, we use stall/rebuffering events. Since it is difficult to compare the number or duration of stall events across different test durations directly, we instead use the metric stall ratio. We define stall ratio as the ratio between the total duration that the video was stalled to the total duration of the video that was played. The duration of the video played is equal to the cut-off length, unless the network throughput was so low that the video playout could not be completed within the 30 minutes of the test. However, this happens very rarely.

5.3 Results

We conduct the testing for two sets of videos 1) 10 YouTube popular videos of different durations 2) 10 YouTube popular videos all of durations greater than 10 minutes. All videos were downloaded from YouTube in the format 1080p DASH MP4 (itag 137).

5.3.1 YouTube Popular Videos

For the first set we used the YouTube channel "Popular on YouTube"[12] with the country chosen as "Worldwide". We picked 10 videos; selecting the first video from each of the categories on the channel that was available in 1080p resolution and that had unrestricted access. The list contains one feature length film as well.

We used maximum sustained throughput values of 2, 3 and 4 Mbps with peak-rates of 3, 4 and 5 Mbps which were selected based on the average bit rates of the videos (4Mbps). Each video is tested ten times for a particular network configuration. We take the trimmed mean of total stall duration for every video, ignoring the highest and lowest values. The final stall ratio is then calculated as the average of the stall ratio for the all the videos. We again see that as the cut-off increases, the stall ratio stabilizes and running the test longer adds little information to the results. When conditions are particularly constrained, as in the case of networks with 2Mbps and 3Mbps, the ratio stabilizes fairly quickly with the curves straightening out already at 30 seconds. However, at 4Mbps, which is very close to the actual required rate for the videos, it takes at least 60 seconds for the results to start stabilizing and up to 120 seconds for it to straighten out.

This scenario better captures the expected results from YouTube active testing since it includes videos of different lengths, including some that are shorter than the cut-off durations.

5.3.2 Videos over 10 minutes

In order to observe the behavior of longer videos, we took a random sample of 10 videos from our itag 137 (DASH 1080p) dataset used in section 4, with the pre-condition that the duration of the video is greater than 10 minutes. These videos have much higher bit rate requirements and so the maximum sustained throughput values used are 14, 15, 16 and 17 Mbps with peak-rates of 15, 16, 17 and 18 Mbps respectively. The videos were again tested multiple times and the final stall ratio is calculated from a trimmed mean as explained previously.

The results are shown in Figure 7, which are similar to our previous observations. Note that when the cut-off is less than 50 seconds, the stall ratio is 0 for the 17Mbps network. Such a result can be misleading when testing network health for YouTube or similar video streaming websites. The y-axis on the two Figures 6 and 7 are different, and this must be taken into account when comparing the curves.

5.4 Discussion

VBR encoded streams can have high bit rate variations depending on the content; one segment of a stream can be very different from another, with very high peaks at some points. We show in this paper that despite this, user-generated Internet video such as YouTube's can be represented by a clip of 1 to 3 minutes sufficiently well. The result is important for active testing in two ways. Firstly, when testing with live services such as YouTube, it is possible to

[12]https://www.youtube.com/channel/UCF0pVplsI8R5kcAqgtoRqoA

gather reliable performance statistics even when tests are cut off prematurely before the video ends. Secondly, when running longer tests the same clip can be joined to form a longer test stream, eliminating the need for maintaining large test files. Since our original dataset is based on charts from 2013, we collected a smaller dataset of about 70 DASH videos (1080p, 720p and 480p) based on YouTube charts of Feb 2015 and observed the same results.

User experience for Internet video is usually measured in terms of start-up delay and the number or duration of rebuffering/stall events. Start-up delay is the amount of time it takes a video to begin playing from the time the user clicks *Play*. DASH videos have more metrics based on how often the client switches quality of the stream and the highest sustained quality. Using a clip from the beginning ensures that the measured start-up delay is unaffected. However, if the test duration is too short, not only stall-related metrics will be misleading, as we showed with our experiment, but DASH related metrics will not be reliable either.

The results can not be extended directly to feature-length films. However, our analysis for feature length films is based on a very small sample and is insufficient. As future work, we propose testing the hypothesis for a larger sample of videos and testing for not only clips from the beginning but also from the middle, as the beginning in case of most films may consist of mostly credits and an opening sequence.

Finally, it is important to note here that our work focuses on the variability of bit rates within a video stream and not the variability of network performance. For instance, conducting longer tests in mobile networks would add more value because it would better capture the variability introduced by the network conditions. This, however, is true regardless of the variations in the media.

6. CONCLUSION

We explored the bit rate variations within an Internet video stream from a measurement perspective. We show statistically that a clip from the beginning of a YouTube video can represent the entire stream and that cutting off video tests prematurely before the video ends, can still provide acceptable user experience metrics. We recommend that the cut-off length should be at least one minute long, although 3 minutes is a more optimal duration. These results are useful in active testing of HTTP streams, which strive to gather reliable performance metrics in short amounts of time in order to minimize overhead traffic and interference with live traffic. Analysis of a small sample of feature-length films indicates the results can not be directly applied to such videos and more detailed analysis is required.

7. ACKNOWLEDGEMENTS

This work was supported by the European Community's Seventh Framework Programme (FP7/2007-2013) grant no. 317647 (Leone) and EC H2020 RIFE project Grant No. 644663.

8. REFERENCES

[1] V. K. Adhikari, S. Jain, and Z.-L. Zhang. Youtube traffic dynamics and its interplay with a tier-1 isp: an isp perspective. In *Proceedings of the 10th ACM SIGCOMM conference on Internet measurement*, pages 431–443. ACM, 2010.

[2] V. K. Adhikari, S. Jain, and Z.-L. Zhang. Where do you" tube"? uncovering youtube server selection strategy. In *Computer Communications and Networks (ICCCN), 2011 Proceedings of 20th International Conference on*, pages 1–6. IEEE, 2011.

[3] S. Ahsan, V. Singh, and J. Ott. Characterizing internet video for large-scale active measurements. *arXiv preprint arXiv:1408.5777*, 2014.

[4] S. Alcock and R. Nelson. Application flow control in youtube video streams. *ACM SIGCOMM Computer Communication Review*, 41(2):24–30, 2011.

[5] A. Brodersen, S. Scellato, and M. Wattenhofer. Youtube around the world: geographic popularity of videos. In *Proceedings of the 21st international conference on World Wide Web*, pages 241–250. ACM, 2012.

[6] P. Casas, R. Schatz, and T. Hoßfeld. Monitoring youtube qoe: Is your mobile network delivering the right experience to your customers? In *Wireless Communications and Networking Conference (WCNC), 2013 IEEE*, pages 1609–1614. IEEE, 2013.

[7] X. Cheng, J. Liu, and C. Dale. Understanding the characteristics of internet short video sharing: A youtube-based measurement study. *IEEE transactions on multimedia*, 15(5):1184–1194, 2013.

[8] Cisco. The Zettabyte Era—Trends and Analysis. Technical Report, 2014.

[9] W. J. Conover. *Practical nonparametric statistics*. Wiley, 1987.

[10] P. D'Urso and E. A. Maharaj. Autocorrelation-based fuzzy clustering of time series. *Fuzzy Sets and Systems*, 160(24):3565–3589, 2009.

[11] P. Eardley, A. Morton, M. Bagnulo, T. Burbridge, P. Aitkin, and A. Akhter. A framework for large-scale measurement platforms (LMAP). draft-ietf-lmap-framework-04, Mar. 2014.

[12] A. Finamore, M. Mellia, M. M. Munafò, R. Torres, and S. G. Rao. Youtube everywhere: Impact of device and infrastructure synergies on user experience. In *Proceedings of the 2011 ACM SIGCOMM conference on Internet measurement conference*, pages 345–360. ACM, 2011.

[13] P. Gill, M. Arlitt, Z. Li, and A. Mahanti. Youtube traffic characterization: a view from the edge. In *Proceedings of the 7th ACM SIGCOMM conference on Internet measurement*, pages 15–28. ACM, 2007.

[14] T. Hoßfeld, M. Seufert, M. Hirth, T. Zinner, P. Tran-Gia, and R. Schatz. Quantification of youtube qoe via crowdsourcing. In *Multimedia (ISM), 2011 IEEE International Symposium on*, pages 494–499. IEEE, 2011.

[15] P. Juluri, L. Plissonneau, Y. Zeng, and D. Medhi. Viewing youtube from a metropolitan area: What do users accessing from residential isps experience? In *Integrated Network Management (IM 2013), 2013 IFIP/IEEE International Symposium on*, pages 589–595. IEEE, 2013.

[16] O. Karkulahti and J. Kangasharju. Youtube revisited: On the importance of correct measurement methodology. In *Traffic Monitoring and Analysis*, pages 17–30. Springer, 2015.

[17] P. Montero and J. A. Vilar. Tsclust: An r package for time series clustering. *Journal of*, 2014.

[18] H. Nam, K.-H. Kim, D. Calin, and H. Schulzrinne. Youslow: a performance analysis tool for adaptive bitrate video streaming. In *Proceedings of the 2014 ACM conference on SIGCOMM*, pages 111–112. ACM, 2014.

[19] L. Plissonneau, E. Biersack, and P. Juluri. Analyzing the impact of youtube delivery policies on user experience. In *Proceedings of the 24th International Teletraffic Congress*, page 28. International Teletraffic Congress, 2012.

[20] R Core Team. *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria, 2015.

[21] SANDVINE. Global internet phenomena report 1h2014. urlhttps://www.sandvine.com/downloads/general/global-internet-phenomena/2014/1h-2014-global-internet-phenomena-report.pdf, 2014.

[22] S. Sonntag, J. Manner, and L. Schulte. Netradar-measuring the wireless world. In *Modeling & Optimization in Mobile, Ad Hoc & Wireless Networks (WiOpt), 2013 11th International Symposium on*, pages 29–34. IEEE, 2013.

[23] G. support. Upload videos longer than 15 minutes, 2010.

[24] G. Tyson, Y. Elkhatib, N. Sastry, and S. Uhlig. Demystifying porn 2.0: a look into a major adult video streaming website. In *Proceedings of the 2013 conference on Internet measurement conference*, pages 417–426. ACM, 2013.