# QoE of YouTube Video Streaming for Current Internet Transport Protocols

Tobias Hoßfeld[1], Raimund Schatz[2], Udo Krieger[3]

[1] University of Würzburg, Institute of Computer Science, Würzburg, Germany
hossfeld@informatik.uni-wuerzburg.de
[2] Telecommunications Research Center Vienna - FTW, Vienna, Austria
schatz@ftw.at
[3] Otto-Friedrich University Bamberg, Bamberg, Germany
udo.krieger@uni-bamberg.de

**Abstract.** Video streaming currently dominates global Internet traffic and will be of even increasing importance in the future. In this paper we assess the impact of the underlying transport protocol on the user perceived quality for video streaming using YouTube as example. In particular, we investigate whether UDP or TCP fits better for Video-on-Demand delivery from the end user's perspective, when the video is transmitted over a bottleneck link. For UDP based streaming, the bottleneck link results in spatial and temporal video artifacts, decreasing the video quality. In contrast, in the case of TCP based streaming, the displayed content itself is not disturbed but playback suffers from stalling due to rebufferung. The results of subjective user studies for both scenarios are analyzed in order to assess the transport protocol influences on Quality of Experience of YouTube. To this end, application-level measurements are conducted for YouTube streaming over a network bottleneck in order to develop models for realistic stalling patterns. Furthermore, mapping functions are derived that accurately describe the relationship between network-level impairments and QoE for both protocols.

**Keywords:** YouTube, Quality of Experience, Stalling, TCP, Loss, UDP

## 1 Introduction

Video streaming dominates global Internet traffic and is expected to account for 57 % of all consumer Internet traffic in 2014 generating over 23 exabytes per month [1]. It can be distinguished between delivery of live video streaming with on-the-fly encoding, like IPTV or Facetime, and delivery of pre-encoded video, so called Video-on-Demand (VoD). The most prominent VoD portal is YouTube which accounts for more than two billion video streams daily [2].

The transport of video streams in the Internet is currently realized either with TCP or UDP. However, due to the diverse features of these protocols their application has a huge impact on the streaming behavior. The usage of TCP guarantees the delivery of undisturbed video content since the protocol itself cares for the retransmissions of corrupted or lost packets. Further, it adapts the transport rate to network congestion, thus minimizing packet loss. If the available bandwidth is lower than the required video bit rate the video transmission lasts longer than the video playback. Thus, the playback is interrupted which is referred to as *stalling*. Hence, in case of TCP the video playback rather than the video itself is disturbed. In contrast, UDP does not perform bandwidth adaptation or guarantee packet delivery, but it transmits the data with the same bit rate as forwarded by the application. Thus, network congestion leads to lost packets which occur as artifacts or jumps in the stream. Hence, the user experiences a *degraded video quality* in terms of visual impairments.

The question arises which transport protocol is more appropriate from the end user's point of view, i.e. the Quality of Experience (QoE). To answer this question we consider a bottleneck scenario in which network capacity is limited. Thus, the available network bandwidth may be lower than the required video bit rate and the user may suffer from stalling and quality degradation for TCP and UDP, respectively. In order to compare the impact of the transport protocols on the QoE, two subjective user studies are presented. In previous work [3], we quantified the impact of stalling on QoE, while [4] executed user surveys to evaluate QoE of video streaming with lost packets.

The contribution of this paper is twofold. First, an intensive YouTube measurement study is conducted in order to quantify the relevant application-level QoS parameters for YouTube over a bottleneck. In particular, the observed stalling patterns are modeled in terms of stalling frequency and stalling length. Second, YouTube video streaming via TCP and via UDP is compared from the end-user perspective by means of subjective user studies [3,4]. The comparison is realized by transforming the results of the subjective tests to the common denominator in the considered scenario, that is the network bandwidth limitation due to the bottleneck. Since [3] provides first a YouTube QoE model for given stalling pattern, the work presented here is the first comparing QoE – and in particular YouTube QoE – for different transport protocols.

The reminder of this paper is structured as follows. Section 2 shows the application-level measurements for YouTube over a bottleneck. This includes the video characteristics in terms of duration and video bit rate as well as the observed stalling patterns which is required to later map the bottleneck bandwidth to QoE. The subjective user study on QoE for YouTube video streaming in the presence of stalling, which means via TCP, is reviewed in Section 3. The QoE model for UDP based transmission of YouTube videos is presented in Section 4. The results of the subjective tests are compared and discussed in Section 5. Finally, Section 6 concludes this work and discusses further research issues.

## 2 Measurement of YouTube Application-Level QoS

In the considered bottleneck scenario for TCP, the available network bandwidth $B$ is limited. When downloading a video which is encoded at a video bit rate $V > B$, stalling may occur. The number $N$ of stallings during the video playout as well as the length $L$ of a single stalling event will both affect the QoE. However, the stalling pattern even in the bottleneck scenario with constant network capacity may be quite complex, since several factors interact and influence the stalling pattern, (a) YouTube's implementation of flow control on application layer [5], (b) TCP's flow control on transport layer, (c) variabe bit rate due to the used video encoding, (d) implementation of the video player and its video buffer.

Therefore, we derive in the following a simple model for the observed stalling patterns based on an application-level measurement study. In Section 2.1, the measurement setup is explained. The observed stalling patterns over the dedicated bottleneck are analyzed in Section 2.2. The notation and variables frequently used throughout this paper are summarized in Table 1.

### 2.1 Setup of Application-Level Measurements

Our YouTube TCP measurement campaign took place from July to August, 2011 during which more than 37,000 YouTube videos were requested, about 35 GByte of data traffic was captured, and more than 1,000 videos were analyzed frame by frame in detail. In addition, 266,245 video descriptions were downloaded from YouTube containing the duration of the videos.

For measuring YouTube video streaming over a bottleneck, the measurement setup included three different components. (1) *Bandwidth shaper.* A network emulation software was used to limit the upload and download bandwidth. In our experiments, the "NetLimiter" bandwidth shaper was applied. (2) *YouTube user simulation.* This component simulated a user watching YouTube videos in his browser. Therefore, a local Apache web server was configured and web pages were dynamically generated, which call the YouTube API for embedding and playing the YouTube video. The embedding of the YouTube videos in an own web page is necessary for monitoring the appliction-level QoS. In order to obtain a random snapshot on YouTube, we randomly searched for videos via the YouTube API and used a public dictionary of english words as keyword for the YouTube search request. (3) *QoS monitor.* The video player status ("playing", "buffering", "ended") and the used buffer size (in terms of number of bytes loaded for the current video) were monitored within the generated web page using Javascript. At the end of the simulation (i.e. when the simulated user completely watched the video, after a certain timeout, or in case of any player errors), the stalling monitoring information and the buffer status were written to a logfile. Further, the network packet traces were captured using wireshark and tshark. As a result, both network-level QoS parameters (from the packet traces) and application-level QoS parameters (the stalling patterns) were captured.

The QoS monitor component provided the data for analyzing the stalling pattern on application level. The YouTube API specifies an event called "onStateChange" which is fired whenever the state of the player changes. For each event, e.g. when the video player switches between buffering of data and playing the video, the current timestamp, the number of bytes loaded, as well as an identifier for the event itself are recorded by the QoS monitor. However, it has

Table 1: Notation and variables frequently used

| | |
|---|---|
| *Variables* | |
| $V$ | total bit rate of video in (kbps) |
| $D$ | duration of video in (s) |
| $B$ | bandwidth limitation in (kbps) |
| $N$ | number of stalling events |
| $L$ | duration of a single stalling event |
| $F$ | stalling frequency $F = {}^{N}/_{D}$ in (1/s) |
| $R$ | packet loss ratio |
| $\rho$ | throughput normalized by video bitrate, i.e. $\rho = {}^{B}/_{V}$ |
| *Functions* | |
| $f_L(N)$ | mapping function between number $N$ of stalling events and MOS values for stalling events of length $L$ via TCP |
| $g_v(R)$ | mapping function between packet loss ratio $R$ and MOS values for videos with resolution $v$ (CIF, 4CIF) via UDP |
| $\Upsilon_L(\rho)$ | mapping function between normalized throughput $\rho$ and MOS values for stalling events of length $L$ via TCP |
| $\Upsilon_v(\rho)$ | mapping function between normalized throughput $\rho$ and MOS values for videos with resolution $v$ (CIF, 4CIF) via UDP |

to be noted that the timer resolution depends on the actual JavaScript implementation within the browser used. In our experiments, we used the Internet explorer within Windows 7 which shows a timer resolution of about 16 ms.

For analyzing the video files, the video contents were extracted from the packet traces. The YouTube API specifies a set of calls for requesting videos via HTTP. Via pattern matching, these HTTP requests and corresponding HTTP objects were identified. YouTube uses DNS translation and URL redirection, as the actual video contents are located on various caching servers, see [6,7,8]. The video contents were then reassembled from the corresponding TCP stream.

The video file itself was parsed by implementing a perl module which analyzed the video frames and extracted meta-information from the video file. As a result, video information like video bit rate, video resolution, used audio and video codecs, or video size and duration were extracted. Furthermore, for each video frame in the video stream, information about the video playback times of frames, the size of the video frames, as well as the type of frames (key frame or interframe) were extracted.

### 2.2 Observed Stalling Patterns over Bottleneck

The aim of this section is to model the observed stalling patterns when the YouTube video is streamed over a bottleneck. The subjective user studies [3] summarized in Section 3 quantify QoE depending on the number $N$ of stalling events and the length $L$ of a single stalling event. A mapping function $f_L(N)$ between the stalling parameters as application-level QoS and the QoE in terms of mean opinion score (MOS) values is provided. Now, we derive the influence of the bottleneck capacity $B$ on the observed stalling pattern in the following. In particular, we depict two exemplary bandwidth limitations, that are $B = 384\,\text{kbps}$ as typical bandwidth of UMTS cell phones and $B = 450\,\text{kbps}$ which is roughly the median of the video bit rate $V$ as observed in our measurement campaign, see the technical report [9] for more details.

**Stalling Frequency.** The stalling frequency $F$ is defined as the ratio of the number of stalling events and the duration $D$ of the video, i.e. $F = N/D$. First, the correlation of $F$ with several influence factors was investigated in terms of Pearson's linear correlation coefficient given in brackets: 1. frame rate (-0.03), 2. video duration (-0.35), 3. median of stalling length (0.37), 4. number of stallings (0.47), 5. mean stalling length ($-0.58$), 6. video bit rate (0.87). Thus, there is no significant correlation between stalling frequency and frame rate, number and length of stalling, or the video duration. The stalling frequency is strongly correlated only with the video bit rate.

Figure 1 depicts the stalling frequency depending on the normalized video demand $x$ for two different bandwidth limitations. The normalized video demand is defined as the ratio of the video bit rate $V$ and the bottleneck capacity $B$, i.e. $x = V/B$. The measurement results for each video clip are plotted with "$\diamond$" marker and "+" marker for $B = 384\,\text{kbps}$ and $B = 450\,\text{kbps}$, respectively. As a result, we see that the measurement results – for both bottleneck capacities – lie in the same area. In particular, the measured frequencies with the corresponding measured video demands can be well fitted by an exponential function which we found by minimizing the least square errors,

$$F(x) = -1.09e^{-1.18x} + 0.36\,. \tag{1}$$

The resulting coefficient of determination of the fitting function $F$ and the measurement data is $D = 0.943$. However, there are several outliers which lie above the dashed line in Figure 1. About 15.22 % of the video clips are assumed to be outliers. We found no statistical correlation between these values of $F$ and any other variables. An in-depth analysis of the packet traces as well as of the video contents did not reveal a clear reason for this. However, we assume that these outliers are caused by the implementation of the video player itself. Considering the correlation coefficients of $F$ and the video bitrate $V$ without the outliers leads to 0.955 and 0.958 for $B = 384\,\text{kbps}$ and $B = 450\,\text{kbps}$, respectively.

Thus, when the bottleneck capacity is equal to the video bit rate, i.e. $x = 1$, the stalling frequency is $F(1) = 0.021$. In that case, a one minute video clip will already stall once due to the variable video bit rate. According to the curve fitting function, the stalling frequency will converge and it is $\lim_{x\to\infty} F(x) = 0.357$. Hence, a one minute clip will stall at most 21 times. However, from QoS perspective, this is not relevant, such high video demands may cause the player to crash anyway. From QoE perspective this is either not relevant, since the user is already annoyed when a few stalling events happen (see Section 3).

**Stalling Length.** Next, we take a closer look at the length $L$ of single stalling events. For each video clip, we measured the durations of each stalling event.

Then, we computed several statistical measures per video clip, including mean and median of the stalling length over the stalling events of an individual clip. However, we found no correlation between the statistical measures of the stalling time and any other variable, i.e. video frame rate, stalling frequency, video bit rate, video duration, number of stallings.

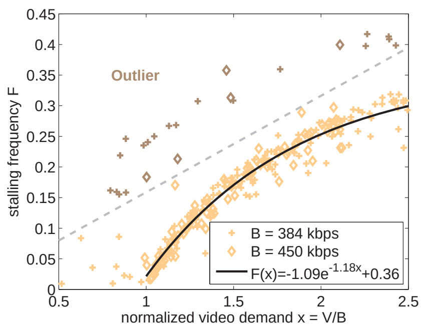



Fig. 1: Measured and fitted stalling frequency $F$ depending on the normalized video demand $x$ as ratio of video bit rate $V$ and bottleneck capacity $B$.

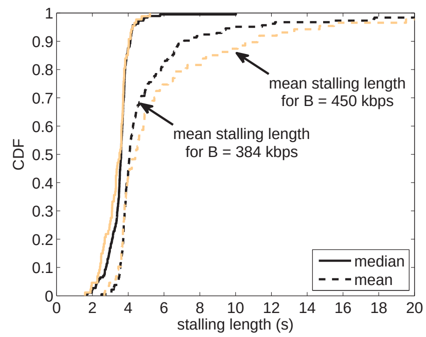



Fig. 2: Median and mean of the stalling length for two different bottleneck capacities of $B = 384\,\text{kbps}$ and $B = 450\,\text{kbps}$, respectively.

Figure 2 shows the CDF of the median and the mean stalling length for the two different network capacities $B$. It can be seen that the curves for the mean stalling length differ with $B$. Nevertheless, the minimum of the average stalling length is about 2 s and for most videos the mean stalling length is below 6 s. However, there are several videos which show an even larger mean stalling length. A closer look at the individual application level stalling traces revealed that this large average stalling length was mostly caused by one large single stalling event during the playout of the individual video clip. These video clips correspond to the outliers as identified for the stalling frequency in Figure 1.

We therefore take a closer look at the median of the stalling length to attenuate the impact of large single stalling events. In that case, the CDFs of the median of the stalling length for the two different network capacities are very close together and no impact of the bottleneck capacity on the median can be observed. In particular, the observed stalling lengths are mainly between 2 s and 5 s. Because of this observation and no correlations with other variables, we conclude that the implementation of the video playout buffer determines mainly the stalling length.

Summarizing this section, the stalling pattern of a video can be described by stalling frequency $F$ and stalling length $L$. The stalling frequency is determined by the ratio of video bit rate and bottleneck capacity. The length of a single stalling event is in the order of a few seconds and lies between 2 s and 6 s mainly.

## 3 Subjective Study on YouTube Video Delivery via TCP

For linking the stalling patterns for YouTube video streaming via TCP to the user perceived quality, we briefly summarize our former subjective user study [3,10] conducted by means of crowdsourcing. Crowdsourcing means to outsource a task (like video quality testing) to a large, anonymous crowd of users in the form of an open call. Crowdsourcing platforms in the Internet, like Amazon Mechanical Turk or Microworkers.com [11], offer access to a large number of internationally widespread users in the Internet and distribute the work submitted by an employer among the users. The work is typically organized at a finer granularity and large jobs (like a QoE test campaign) are split into cheap (micro-)tasks that can be rapidly performed by the crowd.

With crowdsourcing, subjective user studies can be efficiently conducted at low costs with adequate user numbers for obtaining statistically significant QoE scores [12]. However, reliability of results cannot be trusted because of the anonymity and remoteness of participants (cf. [13] and references therein): some subjects may submit incorrect results in order to maximize their income by completing as many tasks as possible; others just may not work correctly due to lack of supervision. In [3,14], we showed that results quality are an inherent problem of crowdsourcing, but can be dramatically improved by filtering based on additional test design measures, e.g. by including consistency and content questions, as well as application usage monitoring.

In several crowdsourcing campaigns, we focused on quantifying the impact of stalling on YouTube QoE and varied (1) the number of stalling events from $N = 0$ to $N = 6$ as well as (2) the length of a single stalling event from $L = 1\,\mathrm{s}$ to $L = 4\,\mathrm{s}$. The stalling events were periodically simulated, i.e. every $D/N$ seconds a single stalling event of constant duration $L$ occured. The duration of all test videos was 30 s. We also considered the influence of (3) the different crowdsourcing campaigns, (4) the test video id in order to take into account the type of video as well as the resolution, used codec settings, etc. Further, we asked the users to additionally rate (5) whether they liked the content.

Table 2: Mapping functions between MOS and number $N$ of stalling events of length $L$ as well as coefficient of determination for TCP transmission

| *length* $L$ | *mapping function* $f_L(N)$ | coef. of determination $R_L^2$ |
|---|---|---|
| 1 s | $f_1(N) = 3.26 \cdot e^{-0.37 \cdot N} + 1.65$ | 0.941 |
| 2 s | $f_2(N) = 2.99 \cdot e^{-0.69 \cdot N} + 1.95$ | 0.923 |
| 3 s | $f_3(N) = 2.99 \cdot e^{-0.96 \cdot N} + 2.01$ | 0.997 |
| 4 s | $f_4(N) = 3.35 \cdot e^{-0.89 \cdot N} + 1.62$ | 0.978 |

As an outcome of this subjective study, we found that the stalling parameters $N$ and $L$ clearly dominate the user ratings and are the key influence factors.

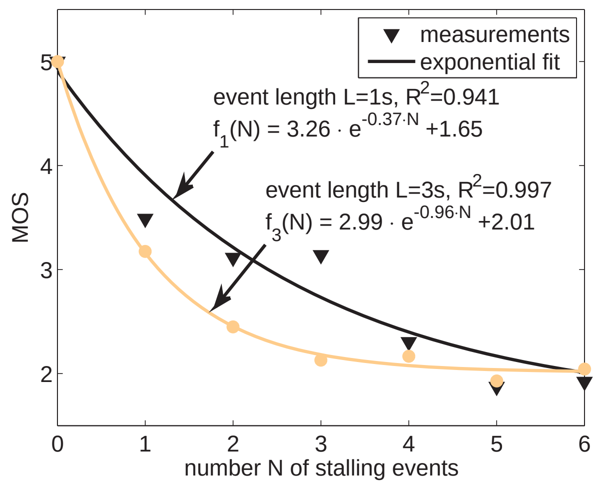



Fig. 3: MOS values for one and three seconds stalling length in case of TCP based video streaming

Surprisingly, the user ratings are statistically independent from the video parameters (like resolution of the YouTube videos, video motion, type of content like news or music clip, etc.) or whether the users liked the content or not.

For quantifying the impact of stalling on QoE, the subjective user ratings for a particular stalling pattern are averaged resulting into a so-called mean opinion score (MOS) according to ITU-T Recommendation ITU-T P.800.1 [15]. MOS takes on the values 1 = bad, 2 = poor, 3 = fair, 4 = good, and 5 = excellent. Figure 3 depicts the MOS values for one and three seconds stalling length for varying number of stalling events. In addition, the MOS values are fitted according the IQX hypothesis as discussed in [16]. The IQX hypothesis formulates a fundamentail relationship between QoE and an impairment factor corresponding to the QoS. According to the IQX hypothesis, the change of QoE depends on the current level of QoE – the expectation level– given the same amount of change of the QoS value. Mathematically, this relationship can be expressed by a differential equation

$$\frac{\partial QoE}{\partial QoS} = -\beta(QoE - \gamma) \tag{2}$$

which can be easily solved as an exponential functional relationship between QoE and QoS.

In the context of YouTube QoE for TCP based video streaming, the number of stallings is considered as impairment. Hence, QoE in terms of MOS is described by an exponential function. The mapping functions between the number $N$ of stalling events of length $L$ are given in Table 2 which also shows the coefficients of determination $R_L^2$ for the different fitting functions being close to perfect match, i.e. $R_L^2 = 1$.

The results in Figure 3 show that users tend to be highly dissatisfied with two ore more stalling events per clip. However, for the case of a stalling length of 1 s, the user ratings are substantially better for same number of stallings. Nonetheless, users are likely to be dissatisfied in case of four or more stalling events, independent of the stalling duration.

## 4 Quality Assessment of UDP-based Video Transmission

For assessing the user perceived quality of YouTube video streaming using the UDP transport protocol, we rely on a publicly available database, that is the "EPFL-PoliMI video quality assessment database" at `http://vqa.como.polimi.it/`. Its video streams are encoded with H.264, the same codec used by YouTube. Twelve different video sequences were investigated from which one half has a spatial CIF resolution ($352\times240$ pixel) and the other half 4CIF resolution ($704\times480$ pixel). For each of the twelve original H.264 bit-streams, a number of corrupted bit-streams were generated, by dropping packets according to a given error pattern. The error patterns were generated at six different packet loss ratios $R$, that are 0.1 %, 0.4 %, 1 %, 3 %, 5 %, 10 %. Furthermore, two different types of error patterns are considered, that are random errors and bursty errors. Thus, in total, 72 CIF and 72 4CIF video sequences with packet losses as well as the original 6 CIF and 6 4CIF sequences without packet losses were considered in the subjective tests.

The CIF and 4CIF video sequences were presented in two separate test sessions to the test users. At the end of each video sequence, the subjects were asked to rate the quality using a five-point ITU continuous adjectival scale. Using a slider, the test users continuously rate the instantaneously perceived quality using an adjectival scale from "bad" to "excellent", which corresponds to an equivalent numerical scale from 0 to 5. Thus, in contrast to the subjective user study in the previous section 3, "bad" quality rating $y$ is any continuous value between 0 and 1, i.e. $0 \leq y \leq 1$, while "excellent" quality rating means $4 < y \leq 5$. In total, fourty naive subjects took part in the subjective tests. More details on the subjective test can be found in [17,4].

Figure 4 shows the MOS depending on the simulated packet loss ratio $R$ for the two different resolutions CIF and 4CIF. For each packet loss ratio $R$ and each video resolution, the subjective ratings from all test users (across the different video contents and the type of error pattern) were averaged to obtain the corresponding MOS value. It can be seen that the MOS strongly decays with increasing network impairment in terms of packet loss.

To this end, we consider the packet loss ratio as impairment factor on the QoE. Hence, we can apply again the IQX hypothesis in order to derive a mapping function between the QoS impairment, i.e. the packet loss ratio, and the QoE in terms of MOS. As a result, we obtain an exponential mapping function between QoE and QoS which is depicted as solid line in Figure 4. Furthermore, the mapping function itself is shown in the plot. Again, we see a very good match of the mapping function and the measured MOS values which is quantified by the coefficient of determination being close to a perfect match.

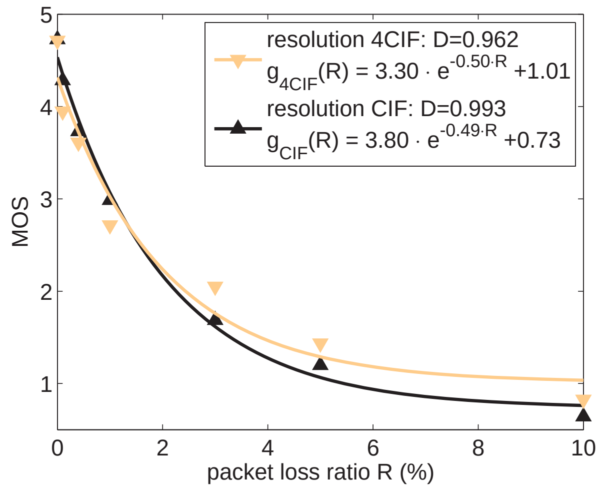



Fig. 4: MOS values and mapping function between packet loss ratio $R$ for UDP based streaming

As a result, we see that in the case of UDP-based video streaming, packet loss is a key influence factor on QoE. In contrast, the resolution of the video contents (CIF vs. 4CIF) has only a minor impact on the MOS.

## 5 Comparison of Youtube QoE for TCP and UDP

For quantifying the influence of the transport protocol on the QoE, we consider now the bottleneck scenario with a given bottleneck capacity $B$. In case of TCP based video streaming, the bottleneck may lead to stalling as QoE impairment. According to our findings in Section 2 a given bottleneck link capacity results in a certain stalling pattern, i.e. a certain stalling frequency $F$ and a certain stalling length $L$. With the YouTube QoE model in Section 3, the stalling pattern can then be mapped to a MOS. In case of UDP based video streaming, the bottleneck link capacity may lead to packet loss as QoE impairment. Then, the QoE model from Section 4 can be applied to quantify the QoE in terms of MOS for a given packet loss ratio $R$. Hence, in both cases, TCP or UDP based video streaming, the bottleneck link capacity is mapped to MOS. In the following, we show how this mapping is applied in case of TCP (Section 5.1) and UDP (Section 5.2). In order to have a fair comparison between UDP and TCP based transmission of video contents, we neglect any initial delays. Finally, Section 5.3 compares both protocols from the end user perspective, when the video stream is delivered over a bottleneck.

### 5.1 TCP based Video Streaming with Stalling

The download time $T_d$ of a video of duration $D$ which is encoded with average video bitrate $V$ depends on the capacity $B$ of the bottleneck,

$$T_d = \frac{V \cdot D}{B}\,. \tag{3}$$

Thus, the total stalling time $T_s$ follows as difference $T_d - D$ between the download time and the video duration,

$$T_s = \left(\frac{V}{B} - 1\right) D\,. \tag{4}$$

Then, the number $N$ of stalling events of length $L$ is

$$N = \left(\frac{V}{B} - 1\right)\frac{D}{L} = \left(\frac{1}{\rho} - 1\right)\frac{D}{L}\,. \tag{5}$$

Together with the normalized throughput $\rho$ which is defined as the ratio between the bandwidth limitation $B$ and the video bitrate $V$, i.e. $\rho = \frac{B}{V}$, we arrive at the following mapping function $\Upsilon_L$ between the normalized throughput and the MOS value,

$$\Upsilon_L(\rho) = f_L\left(\left(\frac{1}{\rho} - 1\right)\frac{D}{L}\right)\,, \tag{6}$$

where $f_L(N)$ is defined as in Section 3 in Figure 3 or Table 2.

In addition to this simple model for obtaining the stalling pattern to a given bottleneck capacity $B$, we can use the fitting function in Eq.(1) which returns the stalling frequency $F = {}^{N}\!/_{D}$ for given ${}^{V}\!/_{B} = {}^{1}\!/_{\rho}$.

### 5.2 UDP based Streaming with Packet Loss

During the video of length $D$, about $\frac{D \cdot B}{S}$ packets of size $S$ are downloaded with a download bandwidth $B$. Since the video (encoded with bitrate $V$) consists of $\frac{D \cdot V}{S}$ packets, the packet loss ratio follows as

$$R = 1 - \frac{B}{V}\,. \tag{7}$$

Accordingly, the mapping $\Upsilon_v$ between the normalized throughput $\rho = \frac{B}{V}$ and the MOS value is derived as

$$\Upsilon_v(\rho) = f_v\,(1 - \rho) \tag{8}$$

using the mapping function $f_v(R)$ between the packet loss ratio $R$ and the MOS value as defined in Section 4 for a given video resolution $v$.

### 5.3 Comparison of QoE for TCP and UDP based delivery of YouTube videos

In this section, we combine the results from the previous subsections in order to compare the QoE for YouTube video streaming over a bottleneck with capacity $B$. For TCP based transmission, this results in stalling which degrades the QoE; for UDP based transmission, the bottleneck results into packet loss and corresponding visual impairments of the video.

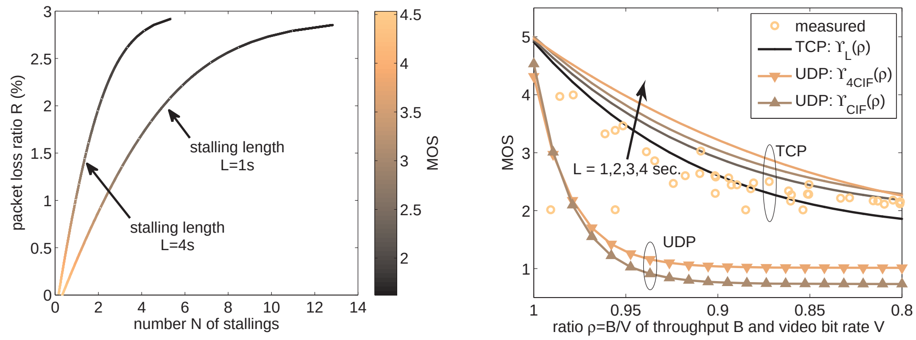
Thus, for the current two Internet protocols, TCP and UDP, the same QoS impairment in terms of the bottleneck bandwidth will lead to completely different QoE impairments. Thus, it is possible to evaluate which kind of stalling pattern (in terms of number of stallings and length of a single stalling event) corresponds to which packet loss ratio, such that the user experiences the same QoE. Figure 5a shows the number $N$ of stallings on the x-axis and the corresponding packet loss ratio $R$ on the y-axis which result in the same MOS value, which is indicated by the color of the point. Two different curves are depicted according to a stalling length of $L = 1\,\mathrm{s}$ and $L = 4\,\mathrm{s}$. For the mapping between packet loss and MOS we used the CIF resolution. For example, $N = 2$ stallings of length $L = 4\,\mathrm{s}$ correspond to a packet loss ratio $R = 2\,\%$ and lead to a MOS value about 2, i.e. bad quality. It can be seen, that the transformation between both impairment factors is quite complex and non-linear.

Finally, we compare both protocols, TCP and UDP, for a given bottleneck bandwidth $B$ in terms of MOS. In particular, we use the normalized throughput $\rho$ as ratio of the bottleneck bandwidth $B$ and the video bitrate $V$. Then, we can directly use the mapping functions in Eq.(6) and in Eq.(8) based on the subjective user studies presented in Section 3 and in Section 4 for TCP and UDP, respectively.

Figure 5b shows the numerical results depending on the normalized throughput $\rho$. In case of TCP, we use the mapping functions based on the four different stalling length from $L = 1\,\mathrm{s}$ to $L = 4\,\mathrm{s}$. In addition, the measurement results from Section 2.2 are used. For the different videos streamed over a bottleneck, we measured the video bitrate, the duration of the video, the observed number of stallings, and the median of the stalling length. These values are used as input in Eq.(6) to obtain a MOS value. The first observation is that the measured stalling values mapped to MOS are in the range of the curves $\Upsilon_L(\rho)$.

In case of UDP, the MOS values are plotted for the CIF and the 4CIF resolution with respect to $\rho$ in Figure 5b. The second observation is that UDP always performs worse than TCP from the end user perspective. Hence, for the same bottleneck capacity, the end user will likely more tolerate the resulting stalling in case of TCP than the resulting video quality degradation in case of UDP.

The results indicate that TCP based video streaming actually used by YouTube outperforms UDP based video streaming in terms of user perceived quality for network bottleneck scenarios. However, it has to be noted that also techniques for overcoming the video quality degradation due to packet losses in case of UDP do exist. By allowing buffering as well as additional retransmission mechanisms on the application layer, UDP based streaming approach might be enhanced




(a) MOS color plot wrt. stalling frequency (TCP) and packet loss ratio (UDP).

(b) Investigation of different bottleneck capacities with videos in CIF resolution.

Fig. 5: Comparison of UDP and TCP streaming in terms of Mean Opinion Scores.

significantly and even keep up with TCP. Furthermore, we have restricted the results of this paper to the bottleneck scenario. Therefore, it would be interesting to investigate if the results can be transfered to lossy links scenarios or if UDP might be the appropriate choice for such scenarios, as the TCP throughput is approximately proportional to $1/\sqrt{R}$, cf. [18]. In addition, an investigation of other transport protocols like DCCP and SCTP would reveal their ability for video streaming and identify the optimal transport protocol for a YouTube like streaming service.

## 6 Conclusions and Future Work

Quality of Experience as a subjective measure of the end-customer's quality perception has become a key concept for analyzing Internet applications like YouTube video streaming from the end user's perspective. Therefore, in this article we have taken a closer look at the impact of the current Internet transport protocols on QoE for YouTube video streaming. In particular, we have investigated the quality degradations which occur in case of network bandwidth bottlenecks in case of TCP and UDP based video streaming.

For UDP based video streaming, a network bottleneck may result into packet loss and therefore visual impairments of the video contents. In contrast, TCP based video streaming, as currently implemented by YouTube, will not suffer from video quality degradation, i.e. the video content itself is not disturbed, however the bottleneck may lead to stalling of the video stream. The question arises which of both protocols is more appropriate in case of a bottleneck from the end user's perspective.

Therefore, we conducted a large-scale measurement study of YouTube video streaming over a bottleneck, in order to derive and model the resulting stalling pattern. This stalling pattern is non-trivial, due to a number of interactions and correlations on several layers of the ISO/OSI stack. However, we found that the stalling patterns can be modeled in the following way: the stalling frequency as ratio of the number of stallings and the video duration simply depends on the normalized video demand, which is the ratio of the video bit rate and the bottleneck link capacity (see 1). However, their relation follows a non-linear exponential function. The median of the length of a single stalling event was found to be between two seconds and four seconds. With these two parameters, the observed stalling pattern can be modeled for a given bottleneck bandwidth.

As second contribution, we presented the results of two subjective user studies from literature and transformed them accordingly in order to predict user perceived quality for a given bottleneck bandwidth. The first subjective measurement campaign considers QoE when stalling occurs in case of TCP video streaming. The second subjective measurement study allows to quantify QoE when packets get lost in case of UDP video streaming. Finally, this allows to compare the influence of UDP and TCP in the bottleneck scenario. Our results show that TCP outperforms UDP for any given bottleneck bandwidth. Furthermore, we have seen that some basic considerations regarding the observed stalling pattern also enable accurate results in terms of predicted QoE.

This work represents an important first step towards the appropriate selection of network protocols and functionality according to the demands and properties of Internet services based on the strict integration of the actual end user's perspective. This QoE optimized selection may be realized by means of functional composition, network virtualization or other frameworks such as the Framework for Internet Innovation [19]. Future work has to deal with application-network interaction in general. For example adaptive streaming [20] may overcome limitations in the network by reducing the application requirements, but adequate QoE models taking into account video quality adaptation have to be derived.

## Acknowledgments

This work was partly funded by Deutsche Forschungsgemeinschaft (DFG) under grants HO 4770/1-1 and TR257/31-1, in the framework of the EU ICT Project SmartenIt (FP7-2012-ICT-317846), the project ACE 2.0 funded by the Austrian competence center program COMET, and the COST QUALINET Action IC1003. The authors alone are responsible for the content.

## References

1. Cisco Systems Inc.: Cisco Visual Networking Index: Forecast and Methodology, 2009-2014. (June 2010)
2. Shiels, M.: YouTube at five- 2 bn views a day . (2011)
3. Hoßfeld, T., Schatz, R., Seufert, M., Hirth, M., Zinner, T., Tran-Gia, P.: Quantification of YouTube QoE via Crowdsourcing. In: IEEE International Workshop on Multimedia Quality of Experience - Modeling, Evaluation, and Directions (MQoE 2011), Dana Point, CA, USA (December 2011)

4. De Simone, F., Tagliasacchi, M., Naccari, M., Tubaro, S., Ebrahimi, T.: H.264/AVC video database for the evaluation of quality metrics. In: Proceedings of the 35th International Conference on Acoustics, Speech, and Signal Processing (ICASSP 2010). (2010)
5. Alcock, S., Nelson, R.: Application flow control in youtube video streams. SIGCOMM Comput. Commun. Rev. **41** (April 2011)
6. Nygren, E., Sitaraman, R.K., Sun, J.: The akamai network: a platform for high-performance internet applications. SIGOPS Oper. Syst. Rev. **44** (August 2010)
7. Mori, T., Kawahara, R., Hasegawa, H., Shimogawa, S.: Characterizing traffic flows originating from large-scale video sharing services. In Ricciato, F., Mellia, M., Biersack, E.W., eds.: Traffic Monitoring and Analysis. Volume 6003 of Lecture Notes in Computer Science., Springer (2010)
8. Adhikari, V., Jain, S., Zhang, Z.: Where do you tube? uncovering youtube server selection strategy. In: IEEE ICCCN 2011. (July 2011)
9. Hoßfeld, T., Zinner, T., Schatz, R., Seufert, M., Tran-Gia, P.: Transport Protocol Influences on YouTube QoE . Technical Report 482, Uni. Würzburg (July 2011)
10. Hoßfeld, T., Schatz, R., Biersack, E., Plissonneau, L.: Internet Video Delivery in YouTube: From Traffic Measurements to Quality of Experience. In Biersack, E., Callegari, C., Matijasevic, M., eds.: Data Traffic Monitoring and Analysis: From measurement, classification and anomaly detection to Quality of experience. Springers Computer Communications and Networks series (2013)
11. Hirth, M., Hoßfeld, T., Tran-Gia, P.: Anatomy of a Crowdsourcing Platform - Using the Example of Microworkers.com. In: Workshop on Future Internet and Next Generation Networks (FINGNet), Seoul, Korea (June 2011)
12. Chen, K., Chang, C., Wu, C., Chang, Y., Lei, C., Sinica, C.: Quadrant of Euphoria: A Crowdsourcing Platform for QoE Assessment. IEEE Network **24**(2) (March 2010)
13. Hirth, M., Hoßfeld, T., Tran-Gia, P.: Cost-Optimal Validation Mechanisms and Cheat-Detection for Crowdsourcing Platforms. In: Workshop on Future Internet and Next Generation Networks, Seoul, Korea (Jun. 2011)
14. Hoßfeld, T., Keimel, C., Hirth, M., Gardlo, B., Habigt, J., Diepold, K., Tran-Gia, P.: CrowdTesting: A Novel Methodology for Subjective User Studies and QoE Evaluation. Technical Report 486, University of Würzburg (February 2013)
15. ITU-T Rec. P.800.1: Mean opinion score (mos) terminology (February 2003)
16. Fiedler, M., Hoßfeld, T., Tran-Gia, P.: A Generic Quantitative Relationship between Quality of Experience and Quality of Service. IEEE Network Special Issue on Improving QoE for Network Services (June 2010)
17. De Simone, F., Naccari, M., Tagliasacchi, M., Dufaux, F., Tubaro, S., Ebrahimi, T.: Subjective assessment of H.264/AVC video sequences transmitted over a noisy channel. In: Proceedings of the First International Workshop on Quality of Multimedia Experience (QoMEX 2009). (2009)
18. Padhye, J., Firoiu, V., Towsley, D., Kurose, J.: Modeling tcp throughput: a simple model and its empirical validation. SIGCOMM Comput. Commun. Rev. **28** (October 1998)
19. Koponen, T., et al.: Architecting for innovation. SIGCOMM Comput. Commun. Rev. **41** (2011)
20. Sieber, C., Hoßfeld, T., Zinner, T., Tran-Gia, P., Timmerer, C.: Implementation and User-centric Comparison of a Novel Adaptation Logic for DASH with SVC. In: IFIP/IEEE International Workshop on Quality of Experience Centric Management (QCMan), Ghent, Belgium (May 2013)