Identifying QoE optimal adaptation of HTTP adaptive streaming based on subjective studies

Tobias Hoßfeld [*], Michael Seufert, Christian Sieber [1], Thomas Zinner, Phuoc Tran-Gia

University of Würzburg, Institute of Computer Science, Würzburg, Germany

1. Introduction

Video distribution networks, like YouTube [1] or Netflix [2], recently adopted HTTP Adaptive Streaming (HAS) technology. HAS allows for a flexible adaptation of the video quality to the available network resources and device capabilities. Thereby, it also mitigates the problem of buffer underruns and the interruption of the playback, i.e., stalling, which is caused by limited network resources.

To apply HAS, the video content has to be available in multiple bit rates, i.e., quality levels, and split into small segments each containing a few seconds of playtime. The client measures the current bandwidth and/or buffer status and requests the next part of the video in an appropriate bit rate such that stalling is avoided and the available bandwidth is best possibly utilized. Hence, the control intelligence, i.e., which segment to stream, has moved from the servers to the clients. The HAS technology is adopted by a wide range of applications and video content providers [3] and is also standardized in ISO/IEC 23009-1 (MPEG-DASH) [4].

Much research in the HAS area tries to find the best adaptation strategy in order to maximize a user's Quality of Experience (QoE). Therefore, HAS adaptation algorithms monitor the current network conditions, as well as video bit rate and buffer status. Based on these monitored data, they decide which quality level to request next in order to avoid stalling to the greatest possible extent. In [5], different adaptation algorithms are compared and classified with respect to user-perceived influence parameters. Such QoE influence parameters of HAS, which are typically investigated, are initial delay, stalling delays and frequencies, played back video quality, and frequency of quality

* Corresponding author at: University of Duisburg-Essen, Chair of Modeling of Adaptive Systems, Schützenbahn 70, 45127 Essen, Germany. Tel.: +49 201 183 7244.

E-mail addresses: tobias.hossfeld@uni-due.de (T. Hoßfeld), seufert@informatik.uni-wuerzburg.de (M. Seufert), c.sieber@tum.de (C. Sieber), zinner@informatik.uni-wuerzburg.de (T. Zinner), trangia@informatik.uni-wuerzburg.de (P. Tran-Gia).

[1] Present address: Technische Universität München, Institute for Communication Networks, Munich, Germany.

switches [6]. However, a holistic QoE model for HAS streaming, which can be used to assess the performance of adaptation algorithms with respect to the user-perceived QoE, is still missing.

In this work, we lay the foundations for benchmarking the performance of HAS adaptation algorithms compared to the theoretical QoE optimum. Therefore, we propose a Mixed Integer Linear Programming (MILP) problem formulation to compute the theoretical optimum for a single client first. Second, subjective crowdsourcing surveys to identify the key influence parameters for HAS streaming are conducted. Based on the subjective results, the appropriate objective function for the MILP is designed. Third, we perform a statistical evaluation based on real network traces for one exemplary video clip. Different adaptation mechanisms from literature are investigated in a test-bed and the achieved QoE is compared with the optimal QoE obtained from MILP. Finally, our approach is extended to a multi-user scenario. If multiple HAS clients share a bottleneck link, like in the case of live streaming, the distributed download control may introduce unfairness with respect to the individual user-perceived qualities. Hence, we investigate whether adaptation algorithms can achieve a fair QoE distribution for multiple clients.

The paper is structured as follows. Section 2 introduces HAS streaming and revisits related work. The evaluation framework used to compute the theoretical optimum is discussed in Section 3. The subjective results on QoE of HAS-based streaming are highlighted in Section 4. Section 5 presents the results for the single user optimization, and Section 6 the results concerning fairness for the IPTV use-case in a multi-user environment. Conclusions are drawn in Section 7.

2. Background and related work

With classical HTTP video streaming, network conditions and video requirements are insufficiently aligned. Either the video bit rate is smaller than the available bandwidth which leads to a smooth playback but spare resources, which could be utilized for a better video quality, or the bit rate is higher than the available bandwidth which introduces delays and will eventually cause stalling (i.e., the interruption of playback due to empty playout buffers), which degrades the Quality of Experience (QoE) severely (e.g., [7,8]). This misalignment is tackled by HTTP Adaptive Streaming (HAS) which is a new technology that improves classical video streaming by flexibly selecting the video quality, which is delivered to the end users.

2.1. Background on HAS technology

HAS requires the video to be available in different bit rates, i.e., in different quality representations, and split into small chunks which contain a few seconds of playtime each. On the client side the current bandwidth condition and/or buffer status are monitored, and the adaptation algorithm decides which part of the video to download next. It requests the next chunk in an appropriate bit rate, such that stalling is avoided and the available bandwidth is best possibly utilized. Quality adaptation can effectively reduce stalling by 80% when bandwidth is decreased under vehicular mobility, and it was responsible for a higher utilization of the available bandwidth when bandwidth increases [9]. Also in non-mobile environments, HAS is beneficial because it avoids stalling by switching the quality when the available bandwidth fluctuates. HAS has several more benefits compared to classical streaming. For example, HAS enables video service providers to adapt the delivered video to the users' demands (e.g., home users vs. mobile users) or to the selected service levels. This allows for flexible pricing schemes which accurately take into account the consumed service levels [10]. Thus, nowadays not only YouTube [11], which is a prominent example, but an increasing number of video applications employ HAS as their default video streaming technology.

2.2. Quality of Experience impact for HAS streaming

In telecommunication networks, the Quality of Service (QoS) is described objectively by network parameters like packet loss, delay, or jitter. However, a good QoS does not necessarily mean that all customers notice the service quality to be good. Thus, Quality of Experience (QoE) was introduced [12], which explicitly refers to subjectively perceived quality by relying on subjective criteria. For classical HTTP video streaming, the key influence factors on QoE are initial delay and stalling [7,13]. HAS can influence both factors by the configured chunk size and trade-off stalling or delay for adaptation (e.g., a small video chunk size leads to less stalling but more quality switches [9,14]). However, it changes the delivered video quality during playback, which introduces an additional impact on the subjectively perceived video quality [8,15].

The adaptation of image quality for layer-encoded videos was investigated in [16], showing that the frequency of switches should be kept as small as possible. If a switch cannot be avoided, its amplitude should be kept as small as possible. Thus, a stepwise reduction of image quality was rated slightly better than one single decrease. Flicker effects for SVC videos, i.e., rapid alternation of base layer and enhancement layer, were analyzed for adaptive video streaming to handheld devices in [17]. As a result, the frequency effect and the amplitude effect were identified, and additionally the influence of content was determined to play a significant role in how adaptation is perceived by the end users. Smooth to abrupt switching of image quality is compared in [18]. Thereby, down-switching is generally considered annoying. Abrupt up-switching, however, might even increase QoE as users might be pleased to notice the visual improvement. A survey on QoE studies on HAS is provided in [6].

Complementary to existing works in literature, we provide a basic QoE model for HAS in Section 4 which returns the QoE optimal playout strategy for any network condition and any video sequence. It has to be noted that the optimization problem can be formulated without quantifying QoE. The results from the conducted QoE, cf. Section 4, indicate the following rationale of the optimization problem. To maximize QoE for a single user, the time the video is played out in its highest quality level should be maximized. If several playout strategies reach the maximum video quality level, then the number of switches should be minimized.

2.3. HAS adaptation algorithms

With detailed knowledge about preconfigured application layer parameters and network conditions it is possible to compute the optimal playout strategy and thus provide an optimal video playout as discussed in Section 3. The HAS adaptation algorithm at an end device, however, lacks detailed knowledge about the current and future network conditions. Based on the current quality indicators on application layer like pre-buffered video length and video quality, and estimations on the current network conditions, e.g, the current TCP congestion window or the average throughput for the last segment, the adaptation algorithm has to decide which segments shall be downloaded next. There are a number of algorithms, each following specific policies when deciding on which chunk to request next. A rate adaptation algorithm based on smoothed bandwidth changes measured through segment fetch time is proposed in [19]. Another approach [20] develops an adaptation engine based on the dynamics of the available throughput in the past and the current buffer level to select the appropriate representation. It is a rather conservative algorithm, which only requests a medium quality level on average but preserves a low switching frequency. In [3], an algorithm for single-layer content of constant bit rate is presented which selects representations according to current bandwidth, current buffer level, and the average bit rate of each segment. A QoE-aware Dynamic Adaptive Streaming over HTTP (DASH) system (QDASH) is presented in [21]. It comprises a quality adaptation algorithm using bandwidth measurements based on packet round-trip times, the current buffer state, and the average fragment size of a quality level to decide what to download next. Further approaches based on control theory are presented in [22,23]. Theses approaches also utilize network and application conditions to provide a smooth video playback. Additionally, they also support multi-server DASH. A very aggressive strategy is presented in [24] which decides only on the current playback buffer which segment to download next. It often delivers the highest quality representation to the end user but also has a very high switching frequency. In [5], the BIEB algorithm is proposed, which downloads segments based on size ratios between the different quality levels. An overview of existing HAS adaptation algorithms and their details is provided in [6].

All existing algorithms select the next segment to download based on technical parameters like bandwidth or video bit rate, but do not take the expected video quality perceived by the end user into account. So far, no model exists which can be used to evaluate the performance of the HAS adaptation algorithms in terms of QoE. A major contribution of this paper is to formulate the optimization problem which allows to investigate any kind of HAS adaptation algorithm and the difference to the QoE optimal solution. In the paper, we exemplary solve this problem for chosen HAS algorithms in a real-test bed. However, the evaluation framework can be applied to compute the efficiency of any HAS adaptation algorithm for arbitrary network scenarios and video characteristics.

3. Framework for evaluation

3.1. Definition of variables and parameters

First of all, the notation and variables frequently used in this work are introduced. A summary can be found in Table 1. It is assumed that U clients are simultaneously in the system who want to stream a video. A video is available in $R = \{1, \ldots, r_{max}\}$ representations and split into n segments. Each segment S_{ij} contains data for τ seconds of the video representation $j \in R$, and has to be played out at time D_i for $i = 1, \ldots, n$. Each user receives an amount of data $V(t) = v$ during the time $[0, t]$. This means, it takes the time $T(v) = V^{-1}(t)$ to download volume v. In compliance with the available download volume, the client downloads segments and plays them out before their respective deadline. After the first segment has been downloaded, the video playout can begin. Any additional time from the start of the video download until the start of the video playback is called start-up/initial delay T_0.

These variables are sufficient to formulate the optimization problems. The Boolean target variable x_{ij} indicates if the client downloads segment S_{ij} or not, and serves as input to the optimization function. Thus, the optimal assignment x_{ij} describes the outcome of an optimal adaptation strategy. This assignment is realizable under the given conditions, however, no indications of the optimal decisions are contained, i.e., the optimal assignment does not indicate when to download which segment.

In order to remove dependencies on the actual bandwidth conditions and video characteristics, the results presented in this work are normalized. Therefore, the bandwidth factor β is introduced. A bandwidth factor $\beta = 1$ means that a video of duration $n\tau$ with total size $S^* = \sum_{i=1}^{n} S_{ir_{max}}$ of the highest quality representation r_{max} can be downloaded completely without stalling and initial delay. In other words, the received download volume at $n\tau$ equals the total size of the highest quality representation, i.e., $V(n\tau) = S^*$.

3.2. Network traffic pattern and video content

As video content we choose "Tears of Steel[2]", an open-source short movie produced and published by the Blender Foundation. The movie has a playback length of about 12 min and features high image quality with fast-paced action scenes and slow-paced character close-ups in a science fiction scenario. We transcoded the movie into H.264/SVC with spatial scalability using the JSVM reference software version 9.19.15 [25]. The GoP (Group of Pictures) size was set to 8 frames, the instantaneous decoding refresh (IDR) period and intra period to 24 frames, and the quantization parameter (QP) was set to 24. A description of the coding parameters can be found in [26]. Three spatial resolutions were configured, 1280×720, 640×360 and

[2] "Tears of Steel" is available at: https://mango.blender.org/.

Table 1
Notations and variables frequently used. Default values are given in square brackets.

Variable	Explanation
U	Number of simultaneous clients in the system
$R[=\{1,2,3\}]$	Available representations
$n[=350]$	Number of segments
$\tau[=2\text{ s}]$	Duration of a segment
S_{ij}	Size of segment i from representation j including all required representations
w_{ij}	Weighting factor indicating the QoE value of segment i for representation j
D_i	Playback deadline for segment i
$T_0[=0\text{ s}]$	Start-up (or initial) delay
$V(t)$	Total amount of data $V(t)$ received by a client during the time $[0,t]$
$T(v)$	Time $T(v)$ required by a client to download volume v; $T(v)$ is the inverse function of $V(t)$, i.e. $T(V(t)) = t$
$x_{ij} \in \{0,1\}$	Target variable indicating if client downloads segment i from representation j ($x_{ij} = 1$) or not ($x_{ij} = 0$)
β	Bandwidth factor for normalization, $\beta = 1 \iff V(n\tau) = \sum_{i=1}^{n} S_{ir_{max}}$

320×180. The encoded movie shows average bitrates of 0.26 Mbps, 0.95 Mbps, and 2.67 Mbps and a maximum bitrate of 1.28, 3.37, and 10.46 for the three spatial layers.

For use with MPEG DASH (Dynamic Adaptive Streaming over HTTP), we chose a segment duration τ of 2 s (48 frames) resulting in $n = 350$ segments in total. Three inter-dependent DASH representations $R = \{1,2,3\}$ from the SVC segments were created by dissecting the SVC bitstream along the spatial scalability. Table 2 shows the properties of each representation r where $r = 1$ corresponds to the lowest quality SVC spatial layer (320×180) and $r = 3$ to the highest (1280×720). Note that scalable video coding is used, which means that for decoding the segment S_{ij}, the segments $S_{i0}, \ldots, S_{i(j-1)}$ are also required. In the following, we define the segment size S_{iz} as the sum of the segment plus all required lower layer segments ($S_{iz} = \sum_{j=1}^{z} S_{ij}$). A total volume of 238.57 MB is required to download the video content in the highest quality, 84.86 MB and 26.52 MB for the medium and lowest quality level, respectively. The DASH segments have an average size from the lowest to the highest layer of 75.77 kB, 242.47 kB, and 681.64 kB with a standard deviation of 37.15 kB, 127.09 kB, and 419.74 kB. The segment sizes of the three representations are depicted in Fig. 1 on a logarithmic scale.

In the evaluation we relay on a realistic traffic pattern recorded in a vehicular mobility scenario by Müller et al. [3]. The traffic pattern was recorded in and around Klagenfurt, Austria driving on a highway while connected to the Internet with a mobile UMTS stick and measuring the throughput of a large HTTP download. The mean measured bandwidth was 359.97 kBps. We adjusted the measured bandwidth over time in such a way, that after $n\tau = 700$ s (i.e., the video duration) the video is completely downloaded in its highest representation ($\beta = 1$), i.e., $V(n\tau) = \sum_{i=1}^{n} S_{iR}$. This results in a mean adjusted bandwidth of 340.82 kBps. The standard deviation of the bandwidth is 174.83 kBps and the lag-1 autocorrelation is 0.89. The network pattern is wrapped around and for each evaluation run a randomized starting point is selected. Thus, different

Table 2
Characteristics of video contents and the segment sizes S_{ir} of representation r.

Representation	$r = 1$	$r = 2$	$r = 3$
Total volume (MB)	26.52	84.86	238.57
Mean segment size (kB)	75.77	242.47	681.64
Maximum segment size (kB)	301.17	789.66	2142.00
Minimum segment size (kB)	3.76	9.60	20.22
Standard deviation (kB)	37.14	127.09	419.74
Coefficient of variation	0.49	0.52	0.62
Lag-1 autocorrelation	0.76	0.82	0.87

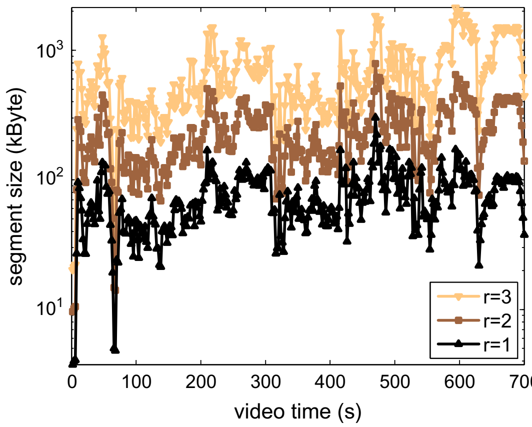


Fig. 1. Segment sizes of the 3 representation layers for the example video of duration 700 s used for the numerical results. The segment sizes are plotted on a logarithmic scale and sum up to 238.57 MB, 84.86 MB, 26.52 MB for $r = 3, 2, 1$.

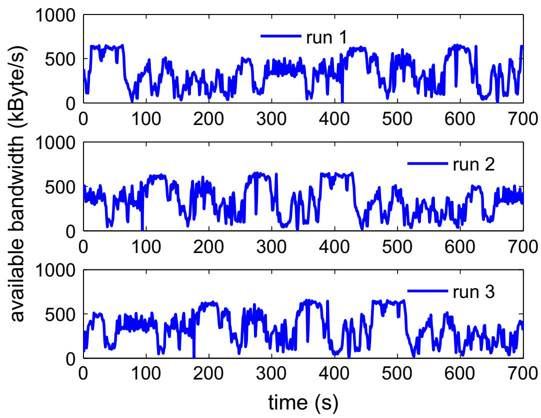


Fig. 2. Network pattern, i.e., available bandwidth over time, of the first three evaluation runs. The measured traffic pattern was adjusted to the given video and wrapped around with a randomized starting point for each evaluation run.

realistic bandwidth patterns can be used albeit statistical characteristics of the bandwidth (e.g., mean, standard deviation, skewness, kurtosis, autocorrelation) are identical in each run. Fig. 2 shows the available bandwidth over time of the first three evaluation runs. It can be seen that the bandwidth fluctuates rapidly in a range from 0.58 kBps to 663.62 kBps during each run.

4. Subjective user study on QoE objectives

For computing the theoretical QoE optimum, we use MILP and formulate a corresponding optimization problem. The objective function of the optimization problem needs to take into account the relevant QoE influence factors. Therefore, it is necessary to understand the main influence factors on HAS QoE as perceived by the end user. To this end, subjective user studies on HAS have been conducted in February 2014 by means of crowdsourcing. The results of the crowdsourcing experiments allow to formulate the rationale of the objective functions as used by the MILP optimization problems.

4.1. Crowdsourcing experiments

In order to have a diverse and large user base for our crowdsourcing experiments, we cooperated with microworkers.com, a large international platform for distributing tasks over the Internet to anonymous workers on the basis of monetary compensation. The platform allows researchers to create a task, define a compensation, and make it available to the crowd. The experiments were set-up utilizing the web-based framework QualityCrowd2 proposed by [27]. The framework allows web-based quality assessment of video content through common web servers and common web browsers on the client side, respectively. To obtain the QoE model for adaptive video streaming, a user study with approximately 100 test subjects was conducted. In the following, we describe the demographics of the crowd and the set-up of the conducted experiment.

Before being able to start the experiment, every participant was asked to complete a short demographic survey. The majority of the users accessed the campaign's web-site from Asia (70%) and from Europe (26%). 42% of the participants were between the age of 22 and 25. The age-groups 18–21 and 26–30 were represented with 18% each. As occupation, 47% of the test subjects specified to be a student, followed by 32% who stated to be in employment. 40% of the participants completed a 4-year college and 17% a 2-year college. 17% stated high school as their highest education. Almost all test persons use the Internet daily (97%) utilizing a fixed line (85% fixed line, 15% mobile access) access technology. A majority of the participants (61%) visit video web-sites several times a day and primarily access the Internet from work (64% at work, 36% at home). 31% of the participants specified to be wearing prescription glasses.

After the demographic survey, a short introduction was presented to the user explaining with pictures how to watch and rate the test sequences. After the user acknowledged the introduction, the test sequences were presented to the participant sequentially. Each test sequence was first completely transferred to the browser cache to prevent any stalling. On completion of the download, a play button was activated for the user to start the playback. After the playback of the video sequence, the user was asked *Did you notice any changes in quality during playback? If yes, did you feel annoyed by them?* and was presented a 5-point ACR slider with the options *Imperceptible (did not notice any)*, *Perceptible but not annoying (did notice, but did not care)*, *Slightly annoying*, *Annoying*, and *Very annoying*.

For the experiment, we choose a 15 s (360 frames) segment of the video content used in the evaluation. The start of the segment corresponds to the timestamp 00:00:25 of the full short-movie. The scene depicts two persons standing on a small bridge and contains a low level of detail and motion, which also results in low spatial/temporal information (SI/TI) values (SI: 8.5, TI: 5.37). We encoded the test sequence in two quality levels by downscaling the source material to 640 × 360 and 160 × 90. Note that in the browser of the user, the two quality levels were both scaled to a window size of 320 × 180.

After the demographic survey, six different quality level switching patterns were presented to the user in random order. Two patterns with zero switches were presented, one which only shows the higher quality to the user and one only showing the lower quality level. The other four patterns start and end on the highest level, but include quality switches which reduce the playout time of the highest quality to 86%, 71%, and 36%, respectively.

4.2. QoE results for HTTP adaptive streaming

The numerical QoE results of the conducted experiments are visualized as bar plot in Fig. 3. It presents the mean opinion scores (MOS) of the different switching patterns, which are ordered along the *x*-axis according to the respective time *t* on the representation layer at highest quality. It can be seen that the user perceived quality for HTTP adaptive streaming is bounded by the quality of highest layer y_H and lowest layer y_L. The bounds y_H and y_L correspond to the mean values of the ratings of the video clip with high and low quality. To be more precise, the MOS values y_H and y_L were obtained in experiments in which the video was played out with constant high and constant low quality, respectively. In Fig. 3, the bounds are plotted with dashed lines. A detailed analysis of the results of the subjective user studies identifies the main influence factors on HAS QoE and assesses the main effect

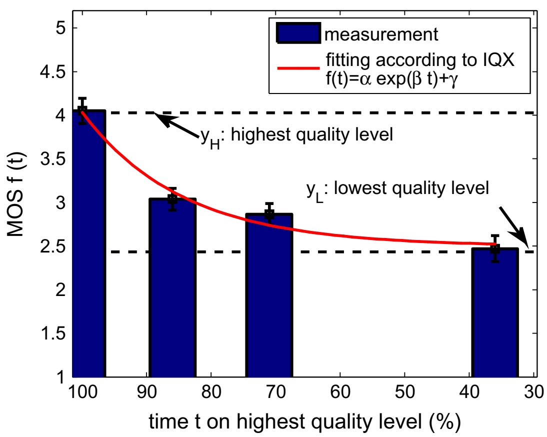


Fig. 3. MOS values of the subjective user study for a video with two representations in high quality level y_H and low quality level y_L. The representations were obtained by using H.264/SVC with different spatial layers to adjust the quality level.

sizes [28]. The results reveal that the time on highest video quality layer is a key influence factor which allows to formulate a simple QoE model.

The model function $f(t)$ maps the time t on the highest layer to the corresponding MOS value. If there is no switch, the equation $f(1) = y_H$ holds, which represents the QoE of the video played out constantly in highest quality. If the video is delivered in low quality level only, $\lim_{t\to 0} f(t) = y_L$ holds. Switching the quality level between the high and low quality level has a negative influence on QoE.

A fundamental functional relationship between QoE and QoS parameters is described by the IQX hypothesis (exponential interdependency of quality of experience and quality of service) [29]. The formula relates changes of QoE with respect to QoS to the current level of QoE and assumes the following differential equation

$$\frac{\partial QoE}{\partial QoS} \sim -(QoE - c) \tag{1}$$

which has an exponential solution. As a result, the IQX hypothesis suggests the following relation f between QoE (in terms of mean opinion scores) and the time t on highest layer:

$$f(t) = ae^{bt} + c. \tag{2}$$

From the MOS values for the different switching patterns, the corresponding fitted function $f(t) = 0.003 \cdot e^{0.064 \cdot t} + 2.498$ can be obtained, which is also plotted in Fig. 3. The fitted function describes the relationship between time on high layer and MOS very well, which is also indicated by a high coefficient of determination $R^2 = 0.98$. It has to be noted that more subjective tests have to be conducted in order to examine additional influence factors and to provide a generic QoE model for HTTP adaptive streaming, e.g., consideration of more than two layers.

Nevertheless, from the observations of the QoE study we conclude the following. To maximize QoE for a single user, the video time played out in its highest quality level should be maximized. This is the basic rationale of the optimization problems formulated in this paper.

It has to be noted that [30] suggests to maximize the downloaded volume which leads to a different quality value function to maximize end user's perception. The rationale behind this assumption is the fact that a representation in a higher quality requires a larger volume than a representation in a lower quality level. However, in practice it may appear that a low quality representation of segment k may be larger than the high quality representation of another segment i, i.e., $S_{i1} > S_{kr},\ r > 1$. In that case, which may be due to different motion patterns and scenes in the video, the optimization would not select the highest possible quality layer. This issue is discussed in more detail in Section 5.3.

5. Optimal adaptation for single user

5.1. Mixed integer linear program for deriving the optimal initial delay

In case of insufficient resources to deliver a video, the video playout buffer may be utilized by delaying the video playout in such a way that the video content can be downloaded without any QoE degradation. In particular, no stalling must occur [31]. Formally, initial delay shifts the regular video segment deadlines, such that the deadline D_i of each segment i can be considered as the sum of the initial delay T_0 and the segment's position $i\tau$ in the video.

$$D_i = T_0 + i\tau, \text{ for all } k = 1, \ldots, n. \tag{3}$$

From the end user's perspective, the objective is to minimize the initial delay [32]. [33] derives a simple closed-form expression for the initial playout buffer level that provides a probabilistic guarantee for undisturbed playback by using a fluid model. We assume however perfect knowledge of $V(t)$ and can therefore derive an optimal initial delay T_0 for compensating insufficient resources when watching the entire video in representation r. For $r = 1$, the obtained initial delay shows the minimum required time in order to achieve smooth playback, while for $r = 3$, the corresponding delay shows the minimum time required to watch the video in its best quality. Before deadline D_i of segment i, the video contents of representation r need to be downloaded completely.

$$\sum_{i=1}^{k} S_{ir} \leqslant V(D_i) = V(T_0 + i\tau), \text{ for all } k = 1, \ldots, n \tag{4}$$

However, Eq. (4) needs to be reformulated as MILP constraint which can be done by using the inverse function $T(v)$ instead of $V(t)$.

$$T_0 \geqslant T(\sum_{i=1}^{k} S_{ir}) - (i-1)\tau, \text{ for all } k = 1, \ldots, n \tag{5}$$

The Optimization Problem 1 formulates the derivation of the optimal initial delay in order to completely download a video in representation r as linear program. Thereby, no segment deadlines must be violated which results in a smooth playback without stalling.

Optimization Problem 1 (Optimal initial delay T_0 for downloading representation r without stalling)

$$\text{minimize} \quad T_0 \in \mathbb{R}_{\geqslant 0} \tag{6}$$

$$\text{subject to} \quad T_0 \geqslant T(\sum_{i=1}^{k} S_{ir}) - (i-1)\tau, \ \forall k = 1, \ldots, n \tag{7}$$

Solving this problem allows to quantify the minimal initial delay which is needed by any algorithm in order to avoid stalling. Fig. 4 shows this optimal initial delay T_0 for different target representations $r \in R$ depending on different bandwidths. The plot is normalized by the bandwidth factor β, which is set to 1 by definition, if the download volume equals the video size in highest representation.

It can be seen that in order to achieve smooth playback without any initial delay the lower quality representations r require a bandwidth factor which is equal to the ratio of the representations' sizes, i.e., $\frac{\sum_i S_{ir}}{\sum_i S_{i3}}$. With lower download volumes, the needed minimum initial delay T_0 increases. Obviously, if the bandwidth factor is cut in half, a user would have to wait a whole playback duration until the video could be played out smoothly.

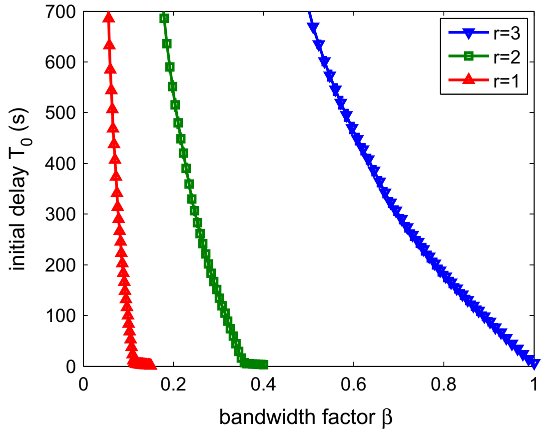


Fig. 4. Optimal initial delay T_0 to download the video contents of representation $r \in \{1, 2, 3\}$ without any stalling of the video playout.

5.2. Optimal adaptation strategy based on objective value functions

A two-step approach for modeling the optimal QoE adaptation for a single user is provided in [30]. The optimal adaptation strategy is formulated and obtained by mixed integer linear programming. In the first step, the downloaded data volume is maximized, since [30] assumes that larger data volume results into higher video quality. In a second step, the number of switches is minimized while stalling is avoided at any time. Based on [30], we use mixed integer linear programing to find the optimal adaptation strategy, but we investigate different objective functions in the first step for maximizing QoE.

For the formulation of the optimization problem, we introduce the target variables $x_{ij} \in \{0, 1\}$ indicating if the client downloads segment i from representation j ($x_{ij} = 1$) or not ($x_{ij} = 0$). The playout of a segment has different impact on QoE depending on the selected representation. Therefore, in order to optimize for QoE, a value function w_{ij} is introduced which indicates the quality value of a segment i in representation j. This value function, which indicates the contribution of a segment to the overall perceived quality, is unknown and has to be determined by future research. In this work, different options for expressing the value of a segment are presented and will be discussed in Section 5.3.

While [30] focused on maximizing the downloaded volume only ($w_{ij} = S_{ij}$), this work investigates whether the proposed optimization problem has to take QoE results into account. In particular, the results from the subjective user studies in Section 4 have shown that the quality layer has to be maximized first. From a practical point of view, it is a natural consequence to minimize the number of switches in a second step in order to avoid flickering affects, which could negatively influence QoE [17]. Thus, two optimization problems 2 and 3 can be formulated. This two-step approach will lead to an optimal QoE without requiring a dedicated QoE model that maps parameters to QoE.

Optimization Problem 2 (Maximize quality value for single user without stalling).

$$\text{maximize} \quad W = \sum_{i=1}^{n}\sum_{j=1}^{r_{max}} w_{ij}x_{ij} \qquad \text{with } x_{ij} \in \{0, 1\} \tag{8}$$

$$\text{subject to} \quad \sum_{j=1}^{r_{max}} x_{ij} = 1 \qquad \forall i = 1, \dots, n \tag{9}$$

$$\sum_{i=1}^{k}\sum_{j=1}^{r_{max}} S_{ij}x_{ij} \leqslant V(D_k) \qquad \forall k = 1, \dots, n \tag{10}$$

This problem will maximize the downloaded quality value depending on the value function w_{ij}. Constraint (9) ensures that for each segment only one representation is downloaded and Eq. (10) ensures that all segments i are downloaded before their deadline D_i. In this respect, $V(D_i)$ represents the maximum amount of data the client can download until the playback deadline of segment i. In the following, the optimal quality value W of Problem 2 will be denoted by W_{opt}.

Optimization Problem 3 (Minimize switches for single user without stalling at given target quality W_{opt})

$$\text{minimize} \quad \frac{1}{2}\sum_{i=1}^{n-1}\sum_{j=1}^{r_{max}} (x_{ij} - x_{i+1,j})^2 \qquad \text{with } x_{ij} \in \{0, 1\} \tag{11}$$

$$\text{subject to} \quad \sum_{j=1}^{r_{max}} x_{ij} = 1 \qquad \forall i = 1, \dots, n \tag{12}$$

$$\sum_{i=1}^{k}\sum_{j=1}^{r_{max}} S_{ij}x_{ij} \leqslant V(D_k) \qquad \forall k = 1, \dots, n \tag{13}$$

$$\sum_{i=1}^{n}\sum_{j=1}^{r_{max}} w_{ij}x_{ij} \geqslant W_{opt} \tag{14}$$

Similarly, constraints (12) and (13) in optimization Problem 3 are the same as constraints (9) and (10) in optimization Problem 2. Additionally, constraint (14) ensures that minimizing the number of quality switches does not decrease the overall quality value below the optimum W_{opt}.

Problem 2 is known as Multiple-Choice Nested Knapsack Problem (MCNKP, [34]), while Problem 3 is a Quadratic MCNKP. It is known that MCNKP is NP-hard, but pseudo-polynomial time algorithms exist which we deploy by using the software gurobi.[3]

5.3. Rationales behind objective value functions

Still the problem remains how to indicate the quality value of a segment. In Table 3, different options for quality value functions are presented. The VOLUME value function resembles the approach of [30] and maximizes the downloaded data volume. The LAYER function weights each segment of representation j by $j \in \{1, \dots, r_{max}\}$ which results in an optimization of the mean representation number.

[3] http://www.gurobi.com/.

Table 3
Different value functions in optimization problems 2 and 3.

Name	Value function	Rationale of objective function
VOLUME	$w_{ij} = S_{ij}$	Maximize downloaded volume, as higher representations need more data volume
LAYER	$w_{ij} = j$	Maximize mean representation
LAYERVOLUME	$w_{ij} = \sum_{k=1}^{n} S_{kj}$	Maximize volume-weighted mean representation
SSIM	$w_{ij} = SSIM_{ij}$	Maximize SSIM metric
HIGHESTLAYER	$w_{ij} = 1,000.00^{j}$, $n < 1,000.00$	Maximize time on highest layer

Similarly, the LAYERVOLUME provides an optimization for the mean representation weighted by the total data volume of layer j. The SSIM function weights each segment by its mean structural similarity (SSIM) index [35]. The HIGHESTLAYER function will always prefer a segment of a higher representation, and thus accounts for an optimization of the time on highest layer.

In order to investigate the different quality value functions they are compared with respect to the achieved maximal average quality level $\bar{l} = \frac{1}{n}\sum_{i=1}^{n}\sum_{j=1}^{3} j \cdot x_{ij}$. Therefore, the optimization problems 2 and 3 were solved for the test video and different bandwidth factors $0.16 \leqslant \beta \leqslant 1$. Smaller bandwidth factors are not meaningful because stalling cannot be avoided in such cases. For each value function, 30 runs were conducted per bandwidth factor with permuted bandwidth patterns as described in Section 3.2. In Fig. 5a, the resulting means and 95% confidence intervals of $\bar{l}$ are plotted. Obviously, the LAYER function optimizes exactly for $\bar{l}$, and thus, results obtained for that function correspond to the best possible results under this metric. However, optimizing the downloaded volume (VOLUME value function) also achieves good results from a QoE point of view and thus could also be considered further. It can be seen that only slightly worse results are reached by using the LAYERVOLUME, HIGHESTLAYER, and SSIM value functions. In any analysis, LAYER and VOLUME perform almost identically, therefore, only LAYER will be considered in the following discussions.

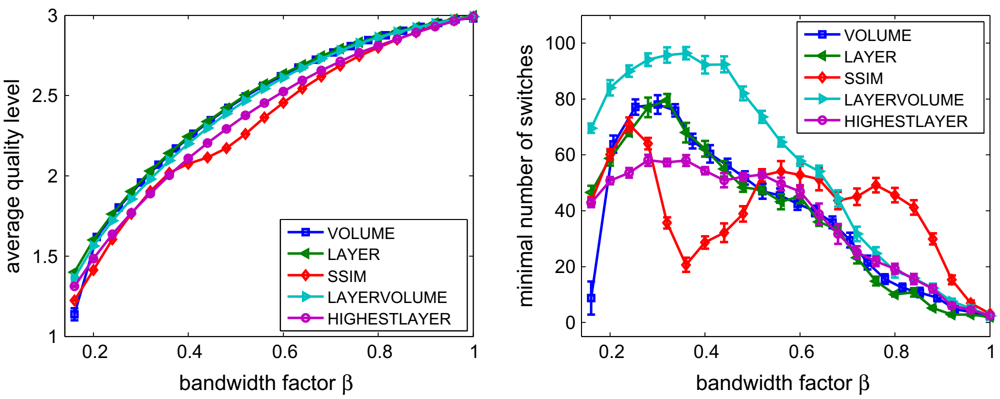
Fig. 5b shows the means and 95% confidence intervals of the minimal number of switches which correspond to the average quality levels in Fig. 5a. It can be seen that SSIM has an early increase of minimal number of switches when the bandwidth factor decreases. With further decreasing bandwidth factor ($\beta < 0.7$), the LAYERVOLUME function accounts for the highest minimal number of switches. The steady gradual increase of LAYER and HIGHESTLAYER is promising for further consideration.

Taking a closer look at what segments are played out for each optimal solution, it can be seen for the LAYER and HIGHESTLAYER value functions that the ratio of highest representation segments increases monotonically when the bandwidth factor increases. The LAYER value function shows a very balanced behavior, as the ratio of lowest quality representation decreases fast and more medium quality ($r = 2$) representations are downloaded. Eventually, with higher β, the number of medium segments decreases again as more highest quality chunks can be downloaded. The HIGHESTLAYER solution, on the other hand, reaches a higher number of highest level ($r = 3$) representations due to its definition, but consists only of lowest and highest quality level segments. As this high switching amplitude results in a lower QoE (cf. [16,17]), the LAYER value function will be considered for the remainder of this work.

5.4. Application for adaptation logic benchmarking

The linear program for optimal adaptation strategies can be used for the performance evaluation of HAS adaptation strategies. Consider an evaluation scenario in which different algorithms are tested for various videos and different network conditions. This allows for a comparison of the algorithms among each other. With the presented linear program, an optimal adaptation strategy can be computed for each video and bandwidth trace. This extends the performance evaluation of adaptation strategies to quantify how close each algorithm reaches the optimum.

As an example, four adaptation algorithms from literature (BIEB [5], Tribler [24], KLU [3], TUB [20]) were



(a) Average quality level $\bar{l}$ (b) Minimal number of switches

Fig. 5. Comparison of optimal solutions for different quality value functions.

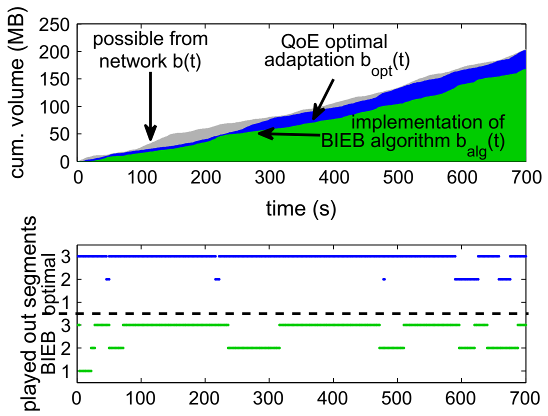


Fig. 6. Single experiment comparing BIEB algorithm with theoretical QoE optimal adaptation strategy for the network scenario sketched in Fig. 2.

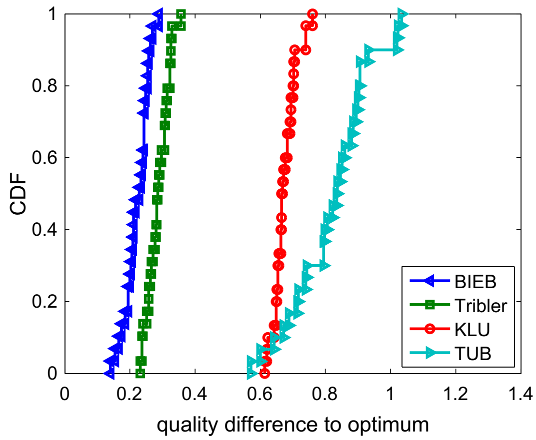


Fig. 7. CDF over 30 simulation runs with different bandwidth patterns comparing algorithm implementations with optimal quality.

compared in a test bed for one video and 30 different bandwidth patterns. Additionally, the linear program was used to compute the optimal strategy for each pattern.

Fig. 6 shows a single experiment for the BIEB algorithm under the network conditions labeled 'run 1' in Fig. 2. To be more precise, the end user is able to download data with bandwidth $b(t)$ which is the available network bandwidth $b(t)$ from the measured traffic trace 'run 1'. The QoE optimal playout strategy only utilizes a fraction of the available bandwidth and downloads the video segments over time with bandwidth $b_{opt}(t) \leqslant b(t)$. Besides the theoretical QoE optimal playout, a concrete implementation of a HAS algorithm, like the BIEB algorithm in Fig. 6, achieves a network utilization below the optimum, as a concrete HAS algorithm does not have knowledge about the current and future network conditions. As a consequence, the HAS algorithm uses a bandwidth $b_{alg}(t) \leqslant b_{opt}(t) \leqslant b(t)$.

In the upper plot of Fig. 6, the x-axis depicts the time of the video playback in seconds, and the y-axis shows the cumulative download volume in MB. The largest area shows the available cumulative download volume $V(t)$ under the given network condition, i.e. the data amount $V(t) = \int_{\tau=t_0}^{t} b(\tau)d\tau$ which can be possibly downloaded over the network from t_0 until t. The area below the largest one depicts the behavior of the theoretical QoE optimal adaptation strategy under the given conditions resulting in the download volume $V_{opt}(t) = \int_{\tau=t_0}^{t} b_{opt}(\tau)d\tau \leqslant V(t)$. The smallest area shows the cumulative download volume of the BIEB algorithm, i.e., the amount of data $V_{alg}(t) = \int_{\tau=t_0}^{t} b_{alg}(\tau)d\tau \leqslant V_{opt}(t)$ that was downloaded by the adaptation logic at the given time t. In the lower plot, the representation $r_{opt}(t)$ and $r_{alg}(t)$ of corresponding played out segments are depicted over the time for both the QoE optimal adaptation and the BIEB implementation, respectively. To be more precise, the plot shows which quality layer from 1 (lowest quality) to 3 (highest quality) was played out at a given time t, respectively.

The illustrative results from the single experiment in Fig. 6 show that the QoE optimal adaptation better utilizes the available bandwidth (especially from around 250 s) because it knows and takes into account the future network conditions. Thus, the optimal strategy is also able to play out a higher quality layer more often than the BIEB algorithm. In addition, it is possible to recognize that the BIEB algorithm does not perform well in the beginning of the video. It plays out layer 1 and 2 segments although download and play out of layer 3 would have been theoretically possible under the given conditions (cf. played out segments by QoE optimal adaptation in the lower part of the figure). These insights gained from the comparison with the QoE optimal adaptation strategy are very valuable for removing the shortcomings of BIEB in future work.

The results of such single experiments can be aggregated for a comprehensive performance evaluation. Fig. 7 shows the CDF of the quality differences of each of the four adaptation algorithms to the optimum over 30 different bandwidth patterns. In general, the algorithms' performance is indicated by the absolute difference to the optimum and the gradient of the CDF. The more left an algorithm is depicted, the closer its performance compared to the optimum. Additionally, the steeper its CDF, the more robust the algorithm with respect to bandwidth fluctuations. It can be seen that the BIEB algorithm outperforms the other investigated algorithms because it is closest to the optimum and shows a robust behavior.

To sum up, with the proposed optimization problems and the corresponding linear program, optimal adaptation strategies can be computed, which indicate what is theoretically possible for any given condition (i.e., video file and bandwidth pattern). This allows for a more comprehensive assessment and benchmarking of the performance of adaptation logics.

6. Multiple users in an IPTV scenario

6.1. Evaluation of shared bottleneck for IPTV

The presented optimization problems can be extended to take into account multiple users. Thereby, it is possible to analyze optimal solutions in case that many users concurrently download and watch the same video over a shared bottleneck link as it is typical for IPTV services. Thus, in the multi-user scenario we consider U different users starting to watch same IPTV content and do a system-wide optimization. Therefore, the optimization variable is extended to x_{uij} to identify which representation j of segment i is downloaded by user u. This allows for the formulation of optimization Problems 4 and 5.

Optimization Problem 4 (Maximize quality value for multi-user scenario without stalling).

$$\text{maximize} \quad W = \sum_{u=1}^{U}\sum_{i=1}^{n}\sum_{j=1}^{r_{max}} w_{ij}x_{uij}, \; x_{uij} \in \{0,1\} \tag{15}$$

$$\text{subject to} \quad \sum_{j=1}^{r_{max}} x_{uij} = 1, \quad \forall u = 1,\ldots,U, \; \forall i = 1,\ldots,n \tag{16}$$

$$\sum_{u=1}^{U}\sum_{i=1}^{k}\sum_{j=1}^{r_{max}} S_{ij}x_{uij} \leqslant V(D_k), \quad \forall u = 1,\ldots,U, \; \forall k = 1,\ldots,n \tag{17}$$

Optimization Problem 5 (Minimize switches for multi-user scenario without stalling at given target quality W_{opt}).

$$\text{minimize} \quad \frac{1}{2}\sum_{u=1}^{U}\sum_{i=1}^{n-1}\sum_{j=1}^{r_{max}} (x_{uij} - x_{ui+1,j})^2, \; x_{uij} \in \{0,1\} \tag{18}$$

$$\text{subject to} \quad \sum_{j=1}^{r_{max}} x_{uij} = 1, \quad \forall u = 1,\ldots,U, \; \forall i = 1,\ldots,n \tag{19}$$

$$\sum_{u=1}^{U}\sum_{i=1}^{k}\sum_{j=1}^{r_{max}} S_{ij}x_{uij} \leqslant V(D_k), \quad \forall u = 1,\ldots,U, \; \forall k = 1,\ldots,n \tag{20}$$

$$\sum_{u=1}^{U}\sum_{i=1}^{n}\sum_{j=1}^{r_{max}} w_{ij}x_{uij} \geqslant W_{opt} \tag{21}$$

Note that there are $U \cdot n$ constraints (cf. Eqs. (16) and (19)), as each user downloads one representation per segment and stalling must be avoided. Thus, the number of runs and the video duration had to be cut due to computing time. Following the considerations from Section 5.3, the LAYER quality value function was used when solving the optimization problems for multiple users. For the evaluation, the average quality levels and minimal numbers of switches of all users are considered as well as fairness aspects.

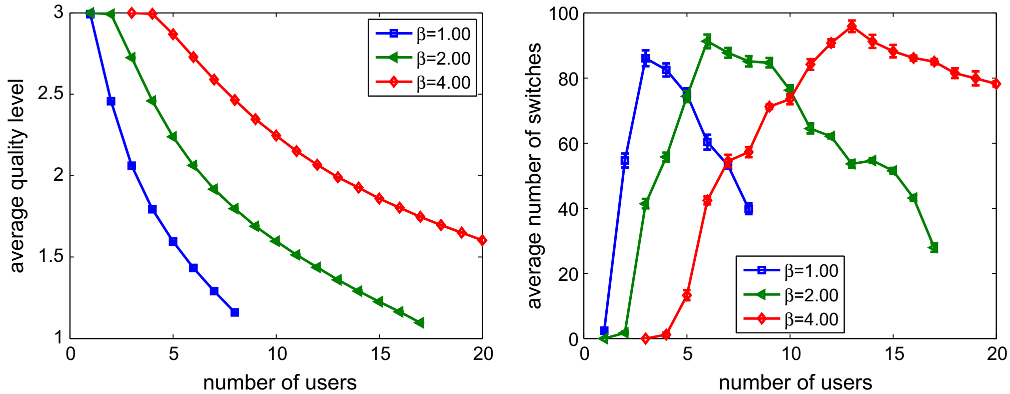
6.2. Impact of service provisioning on QoE and fairness

Fig. 8 presents the mean of the average quality level $\bar{l}$ (a) and the mean of the minimal number of switches and 95% confidence intervals (b) for different number U of users in the system. For $U = 1, 2, 4$, the video can be downloaded and watched in the highest quality if the bandwidth factor β is $1, 2, 4$, respectively, which corresponds to the definition of β. However, the more users are in the system at the same time, the lower average quality levels can be achieved by optimal solutions. In contrast to these rather intuitive results, the minimal number of switches does not follow the same simple principles. It can be observed that it increases rapidly already for few users until it reaches a maximum. This means, that in order to achieve the highest possible average quality level for all users in the system, the optimal solutions rely on an increasing number of quality switches. With more users, $\bar{l}$ drops below 2 and also the number of switches decreases. This is due to the fact that with increasing number of users less representations with $j > 1$ can be downloaded for the optimal solutions. This behavior continues until eventually only representation 1 can be streamed for a maximum number users. If more users would be in the system in parallel, stalling of some users could not be avoided anymore, e.g., only 8 users can be supported in lowest quality for $\beta = 1$ or 17 for $\beta = 2$. It has to be noted that confidence intervals are too small to be visible in these cases.

Fig. 9 relates the results from the multi-user IPTV scenario to a different parameter on the x-axis. In contrast to the number of users as used in Fig. 8, the effective bandwidth β^* is now considered which normalizes the bandwidth factor β by the number U of simultaneous users. Thus, the effective bandwidth is defined as the average bandwidth factor per user, i.e. $\beta^* = \beta/U$. Fig. 9a and b show the average quality level and the average number of switches depending on the effective bandwidth, respectively. Although in the experiments, the bandwidth factor $(\beta = 1, 2, 4)$ as well as the number U of users $(U = 1,\ldots,20)$ are varied, the overlapping curves indicate that both parameters can be abstracted into the effective bandwidth. Thus, the optimal solution in the multi-user IPTV scenario depends only on the effective bandwidth a user obtains as well as the video characteristics.

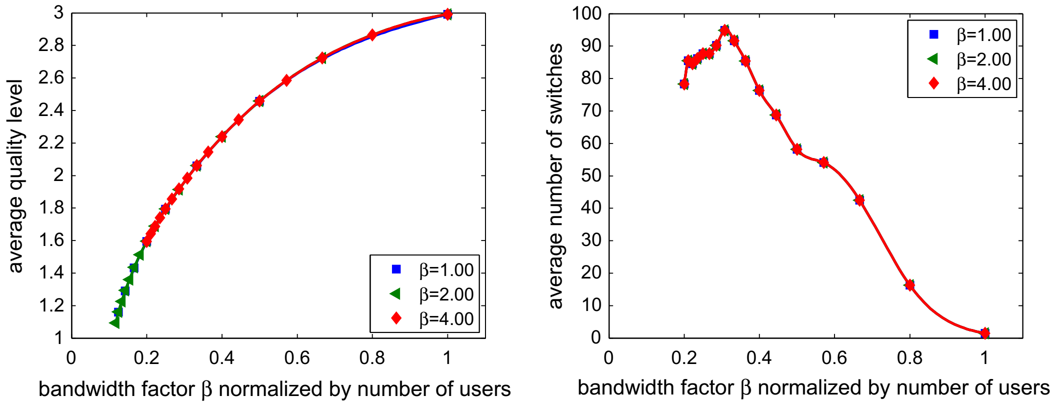
However, these results on their own are not yet meaningful when considering a system-wide perspective. Some users could have to suffer (i.e., download the video in low quality) for the global optimum. Therefore, these optimal solutions are analyzed with respect to their fairness among all users. Jain's fairness index [36] is used, which is defined as $\frac{1}{1+c_x^2}$ with c_x being the coefficient of variation of x (e.g., average quality level). It can be seen that the globally optimal solutions are almost perfectly fair with a fairness index larger than 0.98, i.e., the minimal number of switches is almost the same for all users. The same is true for the average quality level. This means, optimal adaptation strategies in the multi-user scenario are also fair among all users, which was not obvious.

However, [37] revealed that large segment sizes have negative effects on fairness although they allow for a high network utilization. In order to check this finding, the bundling factor b is introduced which means that b segments are bundled into a larger one. Solving the optimization problems for different number of users U and bundling factors b, first of all, no significant impact of U can be found. Thus, exemplary results are shown in the following which can be generalized to all numbers of concurrent users. For $U = 6$, the average quality level reduces only



(a) Average quality level $\bar{l}$ **(b)** Minimal number of switches

Fig. 8. Optimal solution in multi-user IPTV scenario with service provisioning.



(a) Average quality level $\bar{l}$ **(b)** Minimal number of switches

Fig. 9. Effective bandwidth $\beta^* = \beta/U$ per user is considered for the optimal solution in multi-user IPTV scenario with service provisioning for U users. The same data as in Fig. 8 is used, but plotted with the effective bandwidth $\beta^* = \beta/U$ instead of the number U of users on the x-axis.

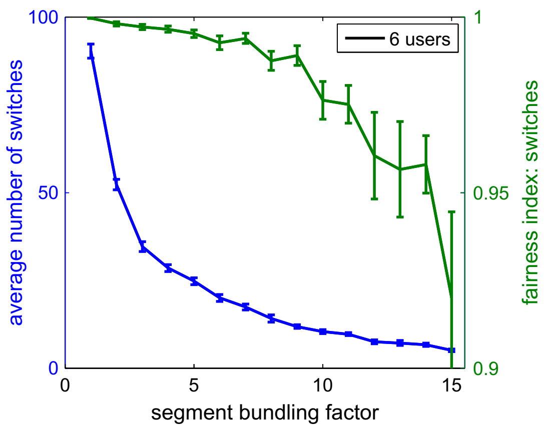


Fig. 10. Influence of the segment bundling factor on the average number of switches and fairness in terms of switches for a multiple user scenario with $U = 6$ users.

minimally from 2.0621 for bundling factor 1 down to 1.9957 for bundling factor 15, and also the fairness index stays very close to 1 (i.e., 0.9991 in the worst case). Fig. 10 shows the impact of different bundling factors b on the minimal number of switches and the resulting fairness index. Evidently, for larger bundling factors, the number of switches is decreased. But also the fairness index in terms of number of switches decreases which means that the number of switches is higher for some users. Thus, it can be confirmed that the optimal solution for larger segment sizes decreases fairness in terms of number of switches but not related to the average quality levels.

7. Conclusions and outlook

HTTP Adaptive Streaming (HAS) provides a more flexible video delivery by allowing end devices to dynamically adjust the video bit rate and therewith the video quality. Multiple downloading strategies have been proposed in literature, which differ with respect to user-perceived application parameters like the average played back quality or the number of quality switches.

The contribution of this paper is threefold. Firstly, we introduced an evaluation framework which allows the computation of theoretical optimum of a HAS downloading algorithm, as well as if QoE fairness in a multiple user environment is possible. Secondly, we performed user surveys to identify the key performance indicators for HAS. It turned out that switching frequently to a better video quality results in a better QoE than keeping a low video quality constantly. Hence, to maximize the overall QoE of a user, the time on highest layer should be maximized, while the number of switches should be minimized. Thirdly, we performed a statistical evaluation of single-user and multi-user scenarios for several downloading strategies. Therefore, we formulated and solved the optimization problems for a set of network conditions and an exemplary video clip. We compared the QoE performance of four existing adaptation strategies to the optimal adaptation and quantified the quality differences. In general, our presented approach allows for a more comprehensive assessment and benchmarking of the performance of adaptation logics with respect to QoE. In the multi-user scenario, we showed that the effective bandwidth per user properly abstracts the network conditions to derive the optimal solution. Based on this we evaluated the fairness among multiple clients competing for a high QoE in case of a shared bottleneck. From a system-wide perspective, the globally optimal solutions indicate a high fairness across the involved users as long as adaption intervals are short. Increasing the length of video segments, however, results in an unfairness in terms of the number of switches while still providing fairness in terms of average played back video quality. As concerns future work, a proper system architecture and a distributed algorithm are to be developed and evaluated which aim at reaching the QoE optimal solution in practice. Dynamic programming techniques for instance may be a promising path to derive novel adaptation strategies providing an optimal video quality without previous knowledge of the currently available networking resources.

Acknowledgments

This work was partly funded by Deutsche Forschungsgemeinschaft (DFG) under grants HO 4770/1-1 and TR257/31-1 and in the framework of the EU ICT Project SmartenIT (FP7-2012-ICT-317846). The authors alone are responsible for the content.

References

[1] B. Rainer, C. Timmerer, Quality of experience of web-based adaptive http streaming clients in real-world environments using crowdsourcing, in: Proceedings of the 2014 Workshop on Design, Quality and Deployment of Adaptive Video Streaming (VideoNext '14), Sydney, Australia, 2014.

[2] M. Ito, R. Antonello, D. Sadok, S. Fernandes, Network level characterization of adaptive streaming over http applications, in: Proceedings of the 2014 IEEE Symposium on Computers and Communication (ISCC), Madeira, Portugal, 2014.

[3] C. Müller, S. Lederer, C. Timmerer, An evaluation of dynamic adaptive streaming over HTTP in vehicular environments, in: Proceedings of the 4th Workshop on Mobile Video (MoVID 2012), Chapel Hill, NC, USA, 2012, pp. 37–42.

[4] International Standards Organization/International Electrotechnical Commission (ISO/IEC), 2012. 23009-1:2012 Information Technology – Dynamic Adaptive Streaming over HTTP (DASH) – Part 1: Media Presentation Description and Segment Formats.

[5] C. Sieber, T. Hoßfeld, T. Zinner, P. Tran-Gia, C. Timmerer, Implementation and user-centric comparison of a novel adaptation logic for DASH with SVC, in: Proceedings of the IFIP/IEEE International Workshop on Quality of Experience Centric Management (QCMan 2013), Ghent, Belgium, 2013, pp. 1318–1323.

[6] M. Seufert, S. Egger, M. Slanina, T. Zinner, T. Hoßfeld, P. Tran-Gia, A survey on quality of experience of HTTP adaptive streaming, IEEE Commun. Surv. Tutor. (2014), http://dx.doi.org/10.1109/COMST.2014.2360940, ISSN: 1553-877X.

[7] T. Hoßfeld, R. Schatz, M. Seufert, M. Hirth, T. Zinner, P. Tran-Gia, Quantification of YouTube QoE via crowdsourcing, in: Proceedings of the IEEE International Workshop on Multimedia Quality of Experience – Modeling, Evaluation, and Directions (MQoE 2011), Dana Point, CA, USA, 2011, pp. 494–499.

[8] O. Oyman, S. Singh, Quality of experience for HTTP adaptive streaming services, IEEE Commun. Mag. 50 (4) (2012) 20–27.

[9] J. Yao, S.S. Kanhere, I. Hossain, M. Hassan, Empirical evaluation of HTTP adaptive streaming under vehicular mobility, in: Proceedings of the 10th International IFIP TC 6 Networking Conference: Networking 2011, Valencia, Spain, 2011, pp. 92–105.

[10] A. Sackl, P. Zwickl, P. Reichl, The trouble with choice: an empirical study to investigate the influence of charging strategies and content selection on QoE, in: Proceedings of the 9th International Conference on Network and Service Management (CNSM), Zurich, Switzerland, 2013, pp. 298–303.

[11] J. Roettgers, Don't Touch That Dial: How YouTube is Bringing Adaptive Streaming to Mobile, TVs, 2013. <http://gigaom.com/2013/03/13/youtube-adaptive-streaming-mobile-tv/>.

[12] P. Le Callet, S. Möller, A. Perkis (Eds.), Qualinet White Paper on Definitions of Quality of Experience (2012), European Network on Quality of Experience in Multimedia Systems and Services (COST Action IC 1003), Lausanne, Switzerland, 2012.

[13] R.K.P. Mok, E.W.W. Chan, X. Luo, R.K.C. Chan, Inferring the QoE of HTTP video streaming from user-viewing activities, in: Proceedings of the ACM SIGCOMM Workshop on Measurements Up the STack (W-MUST), Toronto, ON, Canada, 2011, pp. 31–36.

[14] O. Abboud, T. Zinner, K. Pussep, S. Al-Sabea, R. Steinmetz, On the impact of quality adaptation in SVC-based P2P video-on-demand systems, in: Proceedings of the 2nd Annual ACM Conference on Multimedia Systems (MMSys 2011), Santa Clara, CA, USA, 2011, pp. 223–232.

[15] C. Alberti, D. Renzi, C. Timmerer, C. Mueller, S. Lederer, S. Battista, M. Mattavelli, Automated QoE evaluation of dynamic adaptive streaming over HTTP, in: Proceedings of the 5th International Workshop on Quality of Multimedia Experience (QoMEX 2013), Klagenfurt, Austria, 2013, pp. 58–63.

[16] M. Zink, J. Schmitt, R. Steinmetz, Layer-encoded video in scalable adaptive streaming, IEEE Trans. Multimedia 7 (1) (2005) 75–84.

[17] P. Ni, R. Eg, A. Eichhorn, C. Griwodz, P. Halvorsen, Flicker effects in adaptive video streaming to handheld devices, in: Proceedings of the 19th ACM International Conference on Multimedia (MM 2011), Scottsdale, AZ, USA, 2011, pp. 463–472.

[18] M. Grafl, C. Timmerer, Representation switch smoothing for adaptive HTTP streaming, in: Proceedings of the 4th International Workshop on Perceptual Quality of Systems (PQS 2013), Vienna, Austria, 2013, pp. 178–183.

[19] C. Liu, I. Bouazizi, M. Gabbouj, Rate adaptation for adaptive HTTP streaming, in: Proceedings of the 2nd Annual ACM Conference on Multimedia Systems (MMSys 2011), Santa Clara, CA, USA, 2011, pp. 169–174.

[20] K. Miller, E. Quacchio, G. Gennari, A. Wolisz, Adaptation algorithm for adaptive streaming over HTTP, in: Proceedings of the 19th International Packet Video Workshop (PV 2012), Munich, Germany, 2012, pp. 173–178.

[21] R.K.P. Mok, X. Luo, E.W.W. Chan, R.K.C. Chang, Qdash: a qoe-aware dash system, in: Proceedings of the 3rd Multimedia Systems Conference, MMSys '12, ACM, New York, NY, USA, 2012, pp. 11–22, http://dx.doi.org/10.1145/2155555.2155558. <http://doi.acm.org/10.1145/2155555.2155558>.

[22] G. Tian, Y. Liu, Towards agile and smooth video adaptation in dynamic http streaming, in: Proceedings of the 8th International Conference on Emerging Networking Experiments and Technologies, ACM, 2012, pp. 109–120.

[23] C. Zhou, C.-W. Lin, X. Zhang, Z. Guo, A control-theoretic approach to rate adaptation for dash over multiple content distribution servers, IEEE Trans. Circ. Syst. Video Technol. 24 (4) (2014) 681–694, http://dx.doi.org/10.1109/TCSVT.2013.2290580.

[24] S. Oechsner, T. Zinner, J. Prokopetz, T. Hoßfeld, Supporting scalable video codecs in a P2P video-on-demand streaming system, in:

Proceedings of the 21th ITC Specialist Seminar on Multimedia Applications – Traffic, Performance and QoE (ITC-SS21), Miyazaki, Japan, 2010, pp. 48–53.
[25] Joint Video Team, JSVM Reference Software. <http://www.hhi.fraunhofer.de/en/fields-of-competence/image-processing/research-groups/image-video-coding/svc-extension-of-h264avc/jsvm-reference-software.html>.
[26] I. Unanue, I. Urteaga, R. Husemann, J. Del Ser, V. Roesler, A. Rodríguez, P. Sánchez, A tutorial on h. 264/svc scalable video coding and its tradeoff between quality, coding efficiency and performance, in: Recent Advances on Video Coding 13, 2011.
[27] C. Keimel, J. Habigt, C. Horch, K. Diepold, Qualitycrowd: a framework for crowd-based quality evaluation, in: Picture Coding Symposium (PCS), 2012, IEEE, 2012, pp. 245–248.
[28] T. Hoßfeld, M. Seufert, C. Sieber, T. Zinner, Assessing effect sizes of influence factors towards a QoE model for HTTP adaptive streaming, in: 6th International Workshop on Quality of Multimedia Experience (QoMEX), Singapore, 2014, pp. 1–6.
[29] M. Fiedler, T. Hossfeld, P. Tran-Gia, A generic quantitative relationship between quality of experience and quality of service, IEEE Network 24 (2) (2010) 36–41.
[30] K. Miller, N. Corda, S. Argyropoulos, A. Raake, A. Wolisz, Optimal adaptation trajectories for block-request adaptive video streaming, in: 2013 20th International Packet Video Workshop (PV), IEEE, 2013, pp. 1–8.
[31] T. Hoßfeld, R. Schatz, E. Biersack, L. Plissonneau, Internet video delivery in YouTube: from traffic measurements to quality of experience, in: M.M. Ernst Biersack, Christian Callegari (Eds.), Data Traffic Monitoring and Analysis: From Measurement, Classification and Anomaly Detection to Quality of Experience, Springer's Computer Communications and Networks series, vol. 7754, 2013, pp. 264–301.
[32] T. Hoßfeld, S. Egger, R. Schatz, M. Fiedler, K. Masuch, C. Lorentzen, Initial delay vs. interruptions: between the devil and the deep blue sea, in: QoMEX 2012, Yarra Valley, Australia, 2012, pp. 1–6.
[33] J.W. Bosman, R.D. van der Mei, R. Núñez-Queija, A fluid model analysis of streaming media in the presence of time-varying bandwidth, in: Proceedings of the 24th International Teletraffic Congress, International Teletraffic Congress, 2012, p. 31.
[34] E.Y.-H. Lin, A bibliographical survey on some well-known non-standard knapsack problems, INFOR-Inform. Syst. Oper. Res. 36 (4) (1998) 274–317.
[35] Z. Wang, A.C. Bovik, H.R. Sheikh, E.P. Simoncelli, Image quality assessment: from error visibility to structural similarity, IEEE Trans. Image Process. 13 (4) (2004) 600–612.
[36] R. Jain, D.-M. Chiu, W.R. Hawe, A Quantitative Measure of Fairness and Discrimination for Resource Allocation in Shared Computer System, Eastern Research Laboratory, Digital Equipment Corporation, 1984.
[37] R. Kuschnig, I. Kofler, H. Hellwagner, An evaluation of TCP-based rate-control algorithms for adaptive internet streaming of H.264/SVC, in: Proceedings of the 1st Annual ACM Conference on Multimedia Systems (MMSys 2010), Phoenix, AZ, USA, 2010, pp. 157–168.

Tobias Hoßfeld is professor and head of the Chair "Modeling of Adaptive Systems" at the University of Duisburg-Essen, Germany, since 2014. During the time of this work, he was heading the "Future Internet Applications & Overlays" research group at the Chair of Communication Networks in Würzburg. He finished his Ph.D. in 2009 and his professorial thesis (habilitation) "Modeling and Analysis of Internet Applications and Services" in 2013. He has been visiting senior researcher at FTW in Vienna with a focus on Quality of Experience research. He has published more than 100 research papers in major conferences and journals, receiving 5 best conference paper awards, 3 awards for his Ph.D. thesis, and the Fred W. Ellersick Prize 2013 (IEEE Communications Society) for one of his articles on QoE.

Michael Seufert studied Computer Science, Mathematics, and Education at the University of Würzburg. In 2011, he received his diploma degree in Computer Science, and additionally passed the state examinations which are prerequisites for teaching Mathematics and Computer Science in secondary schools. From 2012–2013, he has been with FTW Vienna working in the area of user-centered interaction and communication economics. Currently, he is a researcher at the University of Würzburg and pursuing his Ph.D. His research mainly focuses on QoE of Internet applications, social networks, performance modeling and analysis, and traffic management solutions.

Christian Sieber received his Master's degree in Computer Science in August 2013 from the Julius-Maximilians Universitä Würzburg (Germany). During his Master's thesis he worked on the subjective and objective evaluation of adaptive video streaming algorithms. In January 2014 he joined the Institute for Communication Networks at the Technische Universität München, where he is now a member of the research staff. Christian is currently working on the configuration management in future communication networks utilizing software defined networking techniques.

Thomas Zinner is heading the NGN research group "Next Generation Networks" at the Chair of Communication Networks in Würzburg. He finished his Ph.D. thesis on "Performance Modeling of QoE-Aware Multipath Video Transmission in the Future Internet" in 2012. His main research interests cover the performance assessment of novel networking technologies, in particular software-defined networking and network function virtualization, as well as network-application interaction mechanisms.

Phuoc Tran-Gia is professor at the Institute of Computer Science and head of the Chair of Communication Networks at the University of Wüurzburg, Germany. His current research areas include architecture and performance analysis of communication systems, and planning and optimization of communication networks. He has published more than 100 research papers major conferences and journals, and recently received the Fred W. Ellersick Prize 2013.