A Survey on Quality of Experience of HTTP Adaptive Streaming

Michael Seufert, Sebastian Egger, Martin Slanina, Thomas Zinner, Tobias Hoßfeld, and Phuoc Tran-Gia

***Abstract*—Changing network conditions pose severe problems to video streaming in the Internet. HTTP adaptive streaming (HAS) is a technology employed by numerous video services that relieves these issues by adapting the video to the current network conditions. It enables service providers to improve resource utilization and Quality of Experience (QoE) by incorporating information from different layers in order to deliver and adapt a video in its best possible quality. Thereby, it allows taking into account end user device capabilities, available video quality levels, current network conditions, and current server load. For end users, the major benefits of HAS compared to classical HTTP video streaming are reduced interruptions of the video playback and higher bandwidth utilization, which both generally result in a higher QoE. Adaptation is possible by changing the frame rate, resolution, or quantization of the video, which can be done with various adaptation strategies and related client- and server-side actions. The technical development of HAS, existing open standardized solutions, but also proprietary solutions are reviewed in this paper as fundamental to derive the QoE influence factors that emerge as a result of adaptation. The main contribution is a comprehensive survey of QoE related works from human computer interaction and networking domains, which are structured according to the QoE impact of video adaptation. To be more precise, subjective studies that cover QoE aspects of adaptation dimensions and strategies are revisited. As a result, QoE influence factors of HAS and corresponding QoE models are identified, but also open issues and conflicting results are discussed. Furthermore, technical influence factors, which are often ignored in the context of HAS, affect perceptual QoE influence factors and are consequently analyzed. This survey gives the reader an overview of the current state of the art and recent developments. At the same time, it targets networking researchers who develop new solutions for HTTP video streaming or assess video streaming from a user centric point of view. Therefore, this paper is a major step toward truly improving HAS.**

This work was supported by the COST Action IC1003 Qualinet, by the Deutsche Forschungsgemeinschaft (DFG) under Grants HO4770/2-1 and TR257/38-1 (DFG project OekoNet), in the framework of the EU ICT Project SmartenIT (FP7-2012-ICT-317846), and by Brno University of Technology under Project CZ.1.07/2.3.00/30.0005.

M. Seufert, T. Zinner, and P. Tran-Gia are with the Institute of Computer Science, University of Würzburg, 97074 Würzburg, Germany (e-mail: seufert@informatik.uni-wuerzburg.de; zinner@informatik.uni-wuerzburg.de; trangia@informatik.uni-wuerzburg.de).

S. Egger was with FTW Telecommunications Research Center, Vienna, Austria. He is now with the Department of Innovation Systems, Technology Experience, AIT Austrian Institute of Technology GmbH, 1220 Vienna, Austria (e-mail: sebastian.egger@aic.ac.at).

M. Slanina is with the Department of Radio Electronics, Brno University of Technology, 616 00 Brno, Czech Republic (e-mail: slaninam@feec.vutbr.cz).

T. Hoßfeld was with the University of Würzburg. He is now with the Chair of Modeling of Adaptive Systems, University of Duisburg-Essen, 45127 Essen, Germany (e-mail: tobias.hossfeld@uni-due.de).

Digital Object Identifier 10.1109/COMST.2014.2360940

***Index Terms*—HTTP adaptive streaming, variable video quality, dynamic streaming, quality of experience.**

I. Introduction

NOWADAYS, video is the most dominant application in the Internet. According to a recent study and forecast [1], global Internet video traffic accounted for 15 PB per month in 2012, which is 57% of all consumer traffic. By 2017, it is expected to reach 52 PB per month, which will then be 69% of the entire consumer Internet traffic. Two thirds of all that traffic will then be delivered by content delivery networks (CDN) like YouTube, which is already today one of the most popular Internet applications.

For a long period of time, YouTube has been employing a server-based streaming, but recently it introduced HTTP adaptive streaming (HAS) [2] as its default delivery/playout method. HAS requires the video to be available in multiple bit rates, i.e., in different quality levels/representations, 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 (i.e., the interruption of playback due to empty playout buffers) is avoided and the available bandwidth is best possibly utilized.

This trend can not only be observed with YouTube, which is a prominent example, but nowadays an increasing number of video applications employ HAS, as it has several more benefits compared to classical streaming. First, offering multiple bit rates of video enables video service providers to adapt the delivered video to the users' demands. As an example, a high bit rate video, which is desired by home users typically enjoying high speed Internet access and big display screens, is not suitable for mobile users with a small display device and slower data access. Second, different service levels and/or pricing schemes can be offered to customers. For example, the customers could select themselves which bit rate level, i.e., which quality level they want to consume. Moreover, adaptive streaming allows for flexible service models, such that a user can increase or decrease the video quality during playback if desired, and can be charged at the end of a viewing session exactly taking into account the consumed service levels [3]. Finally and most important, the current video bit rate, and hence the demanded delivery bandwidth, can be adapted dynamically to changing network and server/CDN conditions. If the video is available in only one bit rate and the conditions change, either the bit rate is smaller than the available bandwidth which leads to a smooth playback but spares resources which could be utilized for a better video quality, or the video bit rate is

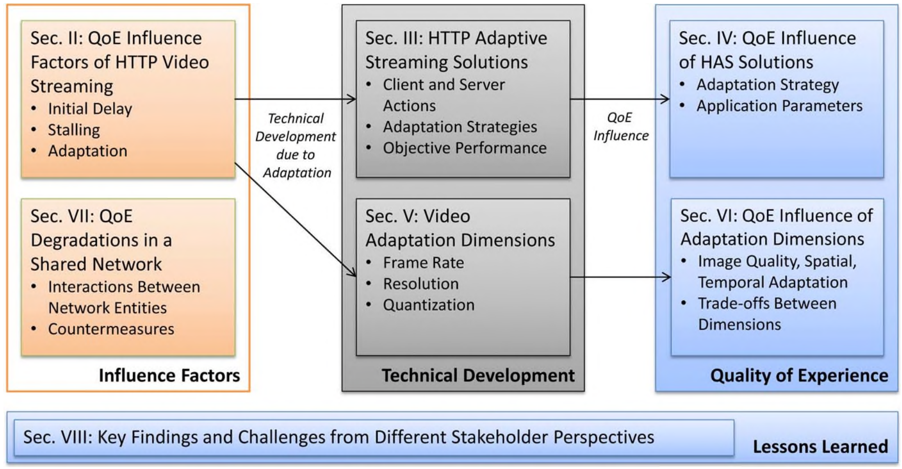

Fig. 1. Structure of the article. Starting from the Quality of Experience of HTTP video streaming, technical possibilities of adaptation and the resulting influence on Quality of Experience are in the focus of this work.

higher than the available bandwidth which leads to delays and eventually stalling, which degrades the Quality of Experience (QoE) severely (e.g., [4] and [5]). Thus, adaptive streaming might improve the QoE of video streaming.

HAS is an evolving technology which is also of interest for the research community. The open questions are manifold and cover both the planning phase and the operational phase.

Planning Phase:

- How (i.e., with which parameters) to convert a source video to given target bit rates?
- Which dimensions (image quality, spatial, temporal) to adapt?

Operational Phase

- When (i.e., under which circumstances) to adapt?
- Which quality representation to request?

Moreover, the performance of existing implementations or proposed algorithms has to be evaluated and improvements have to be identified. This is especially important as today's mechanisms do not take into account the resulting QoE of end users. However, QoE is the most important performance metric as video services are expected to maximize the satisfaction of their users.

In the research community, some surveys exist, which relate to HAS, but mainly focus on visual quality (e.g., [6]–[9]) or streaming technology (e.g., [10]–[14]). However, no survey of the relationship of HAS and subjectively perceived quality has been done. Such QoE aspects of adaptation have already been investigated in subjective end user studies in different disciplines and communities. This work surveys these studies, identifies influence factors and QoE models, and discusses challenges toward new HAS mechanisms. Therefore, this article follows the structure as depicted in Fig. 1: In Section II, the main influence factors of QoE of HTTP video streaming, i.e., initial delay and stalling, are outlined. As changing video quality levels (i.e., adaptation) introduces new impacts on QoE, the remainder of the work will focus on adaptation. Section III presents the approach of HTTP adaptive streaming and describes the current state of the art. Section IV describes the influences on QoE which arise from the employed adaptation strategy and parameters. The dimensions of adaptation which can be utilized for HAS are outlined in Section V. As end users perceive the quality adaptation when using a HAS service, Section VI surveys the influence of each dimension on QoE and describes possible trade-offs. Section VII presents user experience related impairments in a shared network and respective countermeasures. Finally, Section VIII maps the key findings to different stakeholder perspectives discussing the lessons learned, challenges, and future work, and Section IX concludes.

II. Quality of Experience Influence Factors of HTTP Video Streaming

HTTP video streaming (video on demand streaming) is a combination of download and concurrent playback. It transmits the video data to the client via HTTP where it is stored in an application buffer. After a sufficient amount of data has been downloaded (i.e., the video file download does not need to be complete yet), the client can start to play out the video from the buffer. As the video is transmitted over TCP, the client receives an undisturbed copy of the video file. However, there are a number of real world scenarios in which the properties (most importantly instantaneous throughput and latency) of a communication link serving a certain multimedia service are fluctuating. Such changes can typically appear when communicating through a best effort network (e.g., the Internet) where the networking infrastructure is not under control of an operator from end to end, and thus, its performance cannot be guaranteed. Another example is reception of multimedia content through a mobile channel, where the channel conditions are changing over time due to fading, interferences, and noise. These network issues (e.g., packet loss, insufficient bandwidth, delay, and jitter) will decrease the throughput and introduce delays at the application layer. As a consequence, the playout buffer fills more slowly or even depletes. If the buffer is empty, the playback of the video has to be interrupted until enough data for playback continuation has been received. These interruptions are referred to as stalling or rebuffering.

In telecommunication networks, the Quality of Service (QoS) is expressed objectively by network parameters like packet loss, delay, or jitter. However, a good QoS does not guarantee that all customers experience the service to be good. Thus, Quality of Experience (QoE)—a concept of subjectively perceived quality—was introduced [15]. It takes into account how customers perceive the overall value of a service, and thus, relies on subjective criteria. For HTTP video streaming, [4], [16] show in their results that initial delay and stalling are the key influence factors of QoE. However, changing the transmitted video quality as employed by HTTP adaptive streaming introduces a new perceptual dimension. Therefore, in this paper we will present detailed results on the influence of adaptation on subjectively perceived video quality.

In general, the QoE influence factors can be categorized into technical and perceptual influence factors as depicted in Fig. 2. The perceptual influence factors are directly perceived by the end user of the application and are dependent but decoupled from the technical development. For example, several technical reasons can introduce initial delay, but the end user only perceives the waiting time. Therefore, it is necessary to analyze also the technical influence factors, which drive the perception of the end users. In this section, the perceptual influence factors of HTTP video streaming will be described in detail. In later sections (cf. Fig. 1), we will focus on HAS specific parameters.

A. Initial Delay

Initial delay is always present in a multimedia streaming service as a certain amount of data must be transferred before decoding and playback can begin. The practical value of the minimal achievable initial delay thus depends on the available transmission data rate and the encoder settings. Usually, the video playback is delayed more than technically necessary in order to fill the playout buffer with a bigger amount of video playtime in the receiver at first. The playout buffer is an efficient tool used to tackle short term throughput variations. However, the amount of initially buffered playtime needs to be traded off between the actual length of the corresponding delay (more buffered playtime = longer initial delay) and the risk of buffer depletion, i.e., stalling (more buffered playtime = higher robustness to short term throughput variations).

Reference [17] shows that the impact of initial delays strongly depends on the concrete application. Thus, results obtained for other services (e.g., web page load times [18], IPTV channel zapping time [19], and UMTS connection setup time [20]) cannot easily be transferred to video streaming. However, those works presume a logarithmic relationship between waiting times and mean opinion score (MOS), which is a measure of subjectively perceived quality (QoE). References [17] and [21] find fundamental differences between initial delays and stalling. Reference [17] observes that initial delays are preferred to stalling by around 90% of users. The impact of initial delay on perceived quality is small and depends only on its length but not on video clip duration. In contrast to expected initial delay, which is waiting before the service and is well known from everyday usage of video applications, stalling invokes a sudden unexpected interruption within the service. Hence, stalling is processed differently by the human sensory system, i.e., it is perceived much worse [22]. Reference [23]

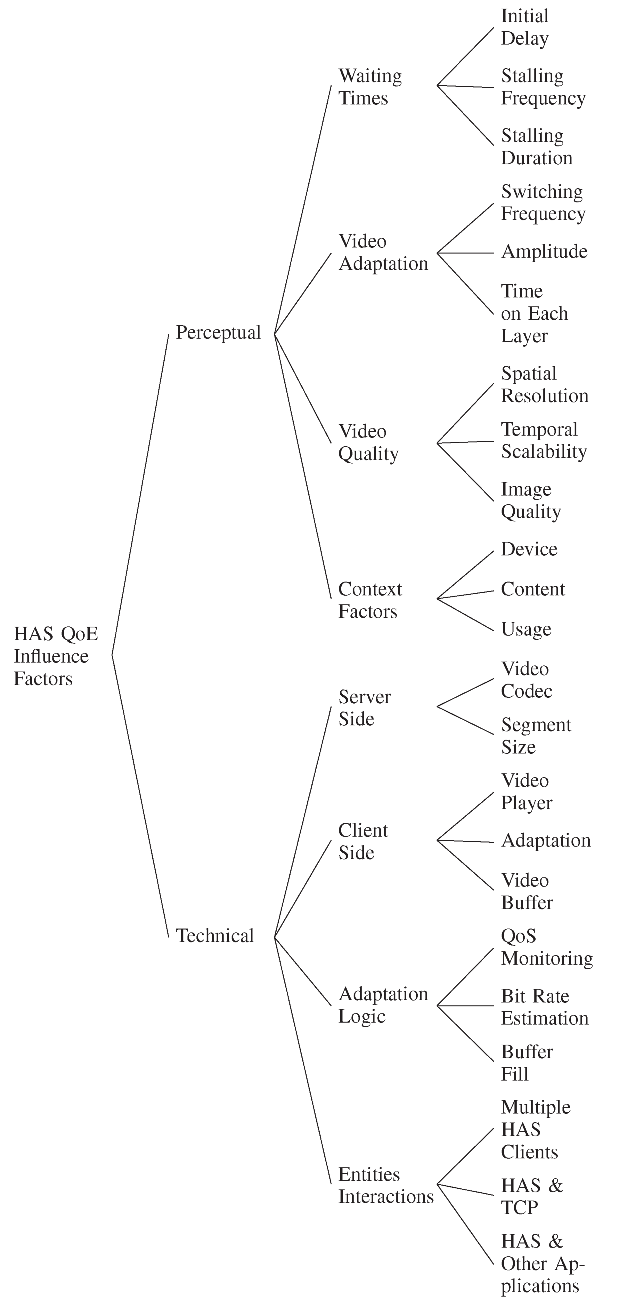

Fig. 2. Taxonomy of HAS QoE influence factors surveyed in this paper. The separation of perceptual and technical influence factors is reflected in the structure of this work.

confirms for mobile video users that initial delays are considered less important than other parameters (e.g., technical quality of the video or stalling) and are less critical for having a good experience.

Thus, the impact of initial delays on QoE of video streaming is not severe. Reference [17] shows that initial delays up to 16 s reduce the perceived quality only marginally. As video service users are used to some delay before the start of the playback, they usually tolerate it if they intend to watch the video. However, recently [24] and [25] describe a new user behavior especially for user-generated contents. They report that users are often browsing through videos, i.e., they start many videos but watch only the first seconds, in order to search for some

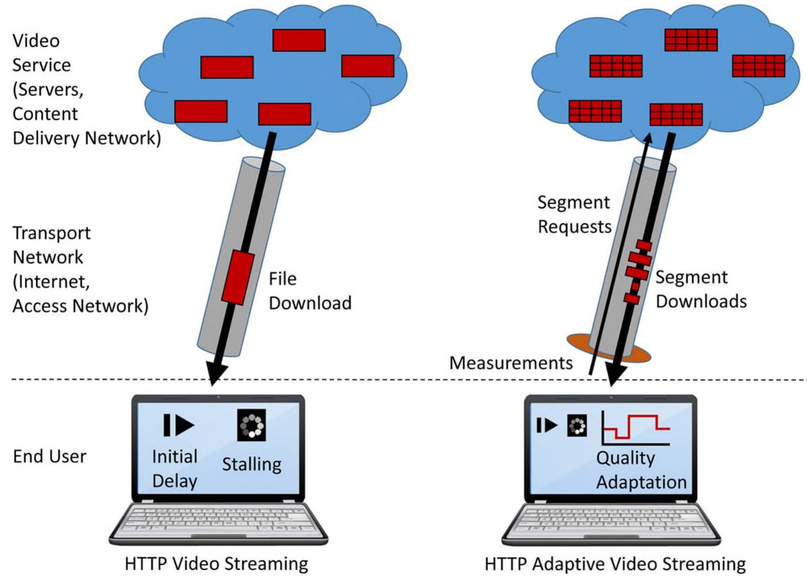

Fig. 3. Comparison of HTTP video streaming and HTTP adaptive video streaming. End user is not aware of service or network influences, but only perceives initial delays, stalling, and quality adaptations.

contents they are interested in. In that case, initial delays should be low to be accepted by the user, however, the QoE related to video browsing is currently not investigated in research yet.

It follows for video service implementations, like for any service, that initial delays should be kept short, but here initial delays are not a major performance issue. Although short delays might be desirable for user-generated content when users often just want to peek into the video, even longer delays up to several seconds will be tolerated, especially if users intend to watch a video.

B. Stalling

Stalling is the stopping of video playback because of playout buffer underrun. If the throughput of the video streaming application is lower than the video bit rate, the playout buffer will deplete. Eventually, insufficient data is available in the buffer and the playback of the video cannot continue. The playback is interrupted until the buffer contains a certain amount of video data. Here again, the amount of rebuffered playtime has to be traded off between the length of the interruption (more buffered playtime = longer stalling duration) and the risk of a shortly recurring stalling event (more buffered playtime = longer playback until potential next stalling event).

In [26], the authors show that an increased duration of stalling decreases the quality. They also find that one long stalling event is preferred to frequent short ones. However, the position of stalling is not important. This last finding is refuted in [27], which shows that there is an impact of the position. In [28], the authors investigate both stalling and frame rate reduction. They show that stalling is worse than frame rate reduction. Furthermore, they show that stalling at irregular intervals is worse than periodic stalling. In [14], stalling is compared to quantization. The authors present a random neural network model to estimate QoE based on both parameters. They find in subjective studies that users are more sensitive to stalling than to an increase of quantization parameter in the video encoder, especially for lower values of the quantization parameter. The authors of [4] present a model for mapping regular stalling patterns to MOS. They show that there is an exponential relationship between stalling parameters and MOS. Moreover, they find that users tolerate at most one stalling event per clip as long as its duration remains in the order of a few seconds. More stalling results in highly dissatisfied users.

Thus, all video streaming services should avoid stalling whenever possible, as already little stalling severely degrades the perceived quality. Classical HTTP video streaming is strongly limited and cannot react to fluctuating network conditions other than by trading off playout buffer size and stalling duration. In contrast, HTTP adaptive streaming is more variable and is able to align the delivered video stream to the current network conditions, thereby mitigating the stalling limitation.

C. Adaptation

HTTP adaptive streaming is based on classical HTTP video streaming but makes it possible to switch the video quality during the playback in order to adapt to the current network conditions. In Fig. 3, both methods are juxtaposed. To make adaptation possible, the streaming paradigm must be changed, such that the client, who can measure his current network conditions at the edge of the network, controls which data rate is suitable for the current conditions. On the server side, the video is split into small segments and each of them is available in different quality levels/representations (which represent different bit rate levels). Based on network measurements, the adaptation algorithm at the client-side requests the next part of the video in the appropriate bit rate level which is best suited under current network conditions.

Reference [29] compares adaptive and non-adaptive streaming under vehicular mobility and reveals that quality adaptation can effectively reduce stalling by 80% when bandwidth decreases, and is responsible for a better utilization of the available bandwidth when bandwidth increases. Also in non-mobile environments, HAS is useful because it avoids stalling to the greatest possible extent by switching the quality. The provisioning–delivery hysteresis presented in [30] confirms that it is better to control the quality than to suffer from uncontrolled effects like stalling. The authors compare video streaming impairments due to packet loss to impairments due to resolution reduction. They map objective results to subjectively perceived quality and find that the impact of the uncontrolled degradation (i.e., packet loss) on QoE is much more severe than the impact of a controlled bandwidth reduction due to resolution.

Thus, HAS is an improvement over classical HTTP video streaming as it aims to minimize uncontrolled impairments. However, compared to classical HTTP video streaming, another dimension, i.e., the quality adaptation, was introduced (see Fig. 3). In the context of HAS, this dimension is not well researched. Therefore, we will present HAS in detail and review subjective studies from different fields on the influences of application layer adaptation on QoE of end users in the following sections. Adaptations on other layers (e.g., network traffic management, modification of content, CDN structure) are not in the focus because end users eventually perceive only resulting initial delays, stalling, or quality adaptations when using a HAS service. Other impairments beyond video QoE, which are caused by usage of HAS in a shared network, are presented in Section VII.

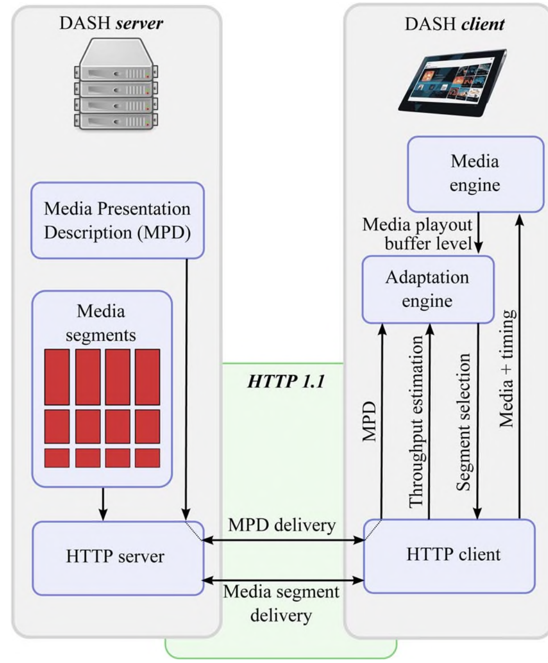

Fig. 4. Principle of adaptive HTTP streaming in a typical DASH system. The server provides media metadata in the MPD, and media segments in different representations. The client requests media segments in desired representations based on MPD information and measurement of throughput and buffer fill level.

III. Current HTTP Adaptive Streaming Solutions

A. *Development and Milestones*

After the first launch of an HTTP adaptive streaming solution by Move Networks in 2006 [31]–[33], HTTP adaptive streaming was commercially rolled out by three dominant companies in parallel—as *Microsoft Silverlight Smooth Streaming* (MSS) [34] by Microsoft Corporation (2008), *HTTP Live Streaming* (HLS) [35] by Apple Inc. (2009) and *Adobe HTTP Dynamic Streaming* (HDS) [36], [37] by Adobe Systems Inc. (2010). Despite their wide adoption and commercial success, these solutions are mutually incompatible, although they share a similar technological background (see Section III-B).

The first standardized approach to adaptive HTTP streaming was published by 3GPP in TS 26.234 Release 9 [38] in 2009 with the intended use in Universal Mobile Telecommunications System-Long Term Evolution (UMTS LTE) mobile communication networks. In the context of [38], the description of the adaptive streaming technique is quite general—the fundamental streaming principle is provided and only a brief description of the media format is given. The work of 3GPP continued by improving the adaptive streaming solution in close collaboration with MPEG [39] and, finally, the Dynamic Adaptive Streaming over HTTP (DASH) standard for general use of HAS was issued by MPEG in 2012 [40]. To date, the DASH specifications are contained in four parts, defining the media presentation description and segment formats, conformance and reference software, implementation guidelines, and segment encryption and authentication, respectively.

Apart from the standardization itself, it is also worth mentioning that in connection with DASH, an industry forum has been formed in order to enable smooth implementation of DASH in different services. Currently, the industry forum groups over 60 members, among which important players in the multimedia and networking market can be found [41]. One of the most important outputs of the industry forum is DASH-AVC/264—a recommendation of profiles and settings serving as guidelines for implementing DASH with H.264/AVC video [42].

B. *Technology Behind*

It has been mentioned in Section III-A that adaptive HTTP streaming solutions, provided as standardized or proprietary technologies by different companies, share a similar technological background. An adaptive HTTP streaming solution architecture can look like the one shown in Fig. 4, in which the terminology used adheres to the DASH specification. Although other HAS solutions use different terminology and different data formats (cf. Tables I and II), the principle of operation is the same.

In a typical HAS streaming session, at first, the client makes a HTTP request to the server in order to obtain metadata of the different audio and video representations available, which is contained in the index file. In DASH, the index file is called Media Presentation Description (MPD, see Fig. 4), while MSS and HDS use the term manifest, and the index file in HLS is called playlist. The purpose of this index file is to provide a list of representations available to the client (e.g., available encoding bit rates, video frame rates, video resolutions, etc.) and a means to formulate HTTP requests for a chosen representation. The most important concept in adaptive HTTP streaming is that switching among different representations can occur at fixed, frequent time instants during the playback, as illustrated in Fig. 5. To achieve this, the media corresponding to the respective representations

TABLE I
COMPARISON OF DIFFERENT Proprietary HTTP ADAPTIVE STREAMING SOLUTIONS. THE DATA DESCRIPTION FORMAT IS EITHER EXTENSIBLE MARKUP LANGUAGE (XML), PLAIN TEXT MULTIMEDIA PLAYLIST (M3U8), OR FLASH MEDIA MANIFEST (F4M). A READER INTERESTED IN AN EXPLANATION OF THE VIDEO AND AUDIO CODECS AND REFERENCES TO THE RESPECTIVE NORMATIVE DOCUMENTS IS REFERRED TO [43]

Proprietary solution	Silverlight Smooth Streaming – MSS	HTTP Live Streaming – HLS	HTTP Dynamic Streaming – HDS
Owner	Microsoft	Apple	Adobe
Data description	Manifest (XML) [44]	Playlist file (M3U8) [35]	Manifest (F4M) [45]
Video codec	H.264, VC-1	H.264	H.264, VP6
Audio codec	AAC, WMA	AAC (HE, LC), MP3, AC3	MP3, AAC
Format	fMP4 [34] *.ismv + *.isma files	M2TS [35] *.ts files	fMP4 [37] *.f4f files
Segment length (typical)*	2 s [34]	10 s [35]**	2 – 5 s [37], [46]***

* No distinction between live and on demand playback is done in the cited documents.
** Shorter segments result in more frequent refreshes of the index file, introducing extra network overhead.
*** 2–3 seconds recommended for content up to 30 seconds long with many scene changes.

TABLE II
COMPARISON OF DIFFERENT Standard HTTP ADAPTIVE STREAMING SOLUTIONS

Standard solution	MPEG DASH [40]	3GP DASH [38]	HbbTV DASH [47]
Owner	Standard	Standard	Recommendation
Data description	Media Presentation Description (XML)	Media Presentation Description (XML)	Media Presentation Description (XML)
Video codec	any	H.264	H.264
Audio codec	any	Enhanced aacPlus, AAC-LC, AAC-LTP Extended AMR-WB	HE-AAC, E-AC3
Format	MP4 or M2TS	3GPP File Format	MP4
Segment length (typical)	not specified	not specified	1 – 15 s

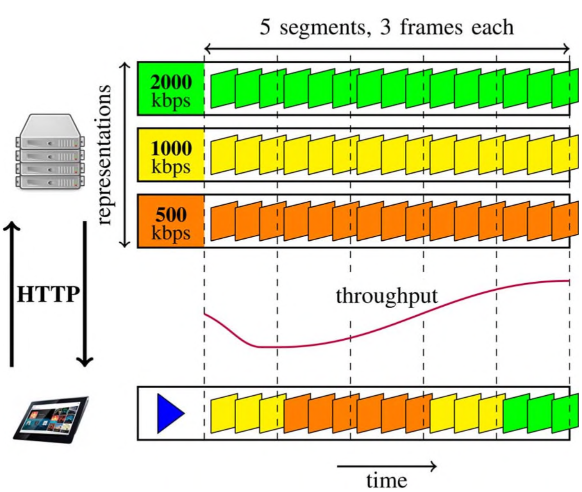

Fig. 5. Video representations available in adaptive HTTP streaming. The HAS server stores the video content encoded in different representations, each representation is characterized by its video bit rate, eventually also resolution, frame rate, etc. (not displayed here). The vertical dashed lines represent the points at which the client can switch among different representations. The switch points are at the boundaries of neighboring segments and are frame aligned in all the representations available. Switching is controlled by the adaptation engine (see Fig. 4) based on the estimation of server-client link throughput (red curve).

is split up into parts of short durations (segments or chunks) typically 1 to 15 s long and either stored on the server as one file per segment (e.g., HLS) or extracted from a single file at run-time based on the client's request (e.g., DASH). The adaptation engine in the client decides which of the media segments should be downloaded based on their availability (indicated by the index file), the actual network conditions (measured or estimated throughput), and media playout conditions (playout buffer fill level). To allow for smooth switching among different representations, the segments corresponding to different representations must be perfectly time (frame) aligned.

The application control loop is shown in the upper part of Fig. 6. Based on the measurement of relevant parameters (e.g., available bandwidth or receiver buffer fullness—more details are discussed in Section III-D), the client's decision engine selects which representation to download next. In this work, the main focus is on the decisions of a single HAS instance and their impacts on QoE. However, there is a complex interplay of the control loop with other applications and the network which can also affect QoE. Therefore, the interactions between different HAS players, other applications, and the interactions with the TCP congestion control loop are discussed in Section VII.

Although the different HAS solutions share the basic principle as illustrated in Fig. 4, they differ in a number of technical parameters. The main features of the proprietary and standard HAS solutions are summarized in Tables I and II, respectively. In the context of this paper, the following parameters are of high importance:

Codec: Although we can find codec agnostic (MPEG DASH) or variable codec (MSS, HDS) solutions, several systems are

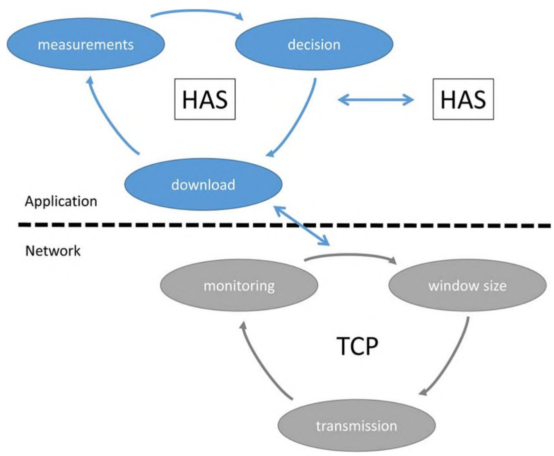

Fig. 6. Control loop of HAS. Based on measurements the client decides which segment to download next. The control loop interacts with other applications (e.g., other HAS instances) and the network (e.g., TCP congestion control loop) which can also affect QoE.

tailored to specific supported coding algorithms for both video and audio. Currently there is an obvious domination of H.264 for video, which is supported by all solutions in our scope, however it can be expected that the recently standardized H.265/HEVC (High Efficiency Video Coding) [48] will take part of its share in the coming years. For audio, the codec selection flexibility is generally higher. The implications of audio and video encoder selection and configuration are further discussed in Section III-C.

Format: The currently dominant formats for media encapsulation are the MPEG-2 transport stream (M2TS [49]; used by HLS, DASH) and ISO Base Media File Format (MP4 [50]; used by DASH) or its derivative referred to as fragmented MP4 (fMP4 [51], [52]; used by HDS and MSS).

MPEG-2 transport streams carry the data in packets of a fixed 184-byte length plus a 4-byte header. Each packet contains only one type of content (audio, video, data, or auxiliary information). M2TS is commonly used for streaming, although its structure is not as flexible as in the case of MP4. MP4 organizes the audiovisual data in so-called boxes and treats different types of data separately. Further, there is a high flexibility in handling the auxiliary information (such as codec settings), which can be tailored to the needs of the application. As such, MP4 has been successfully adjusted to carry prioritized bitstreams of scalable video over HAS [53], which we further consider in Section III-C.

Segment Length: The segment length used in a HAS system specifies the shortest video duration after which a quality (bitrate) adjustment can occur. Although some systems keep these values fixed (Table I), the segment length can be left up to the individual implementation in many cases (Table II). Again, we will further discuss the implications related to segment length design in Section III-C.

It is clear from the principle of adaptive HTTP streaming that the decision engine responsible for selecting appropriate representations is running on the client side and needs to select the representation to request based on different criteria. These criteria can be measurements of downlink throughput, the actual video buffer status, device or screen properties, or context information (e.g., mobility). On the server side, in contrast, the most important decisions are done regarding the preparation of the content, i.e., what representations shall be provided, and regarding its delivery, e.g., the selection of the best CDN server for each request.

The behavior of the HAS system needs to obey the requirements of the actual use case. For example, in the context of a live system, the content is made available at the server during the viewing and a low overall delay introduced by the system shall be achieved. This implies that the provided segment lengths should be small and the segments need to start downloading as soon as they appear on the server. For video on demand systems, on the contrary, a larger receiver buffer can be used together with longer segments to avoid flickering caused by frequent quality representation changes. The following sections discuss the actions the system or its designer can take on the server side and on the client side in order to efficiently adapt to the actual conditions while considering context requirements.

C. Server-Side Actions

On the server side of an HTTP adaptive streaming system, the main concern is the preparation of the content, i.e., selection of available representations, and optimal encoding. This also includes proper selection of system parameters, such as the length of an encoded segment (where selectable).

The length of the video segment needs to obey two contradictory requirements. First, the segments need to be short enough to allow for fast reaction to changing network conditions. On the other hand, the segments need to be long enough to allow high coding efficiency of the source video encoder [54] and to keep the amount of overhead low (the impact of segment size on the necessary overhead is quantified in [55]). Clearly, these two requirements form an optimization problem which needs to be considered at the server side during content preparation.

In [56], the length of video segments to be offered to the client is optimized based on the content, so that I-frames and representation switches are placed at optimal positions, e.g., video cuts. Such an approach led to approximately 10% decrease of the required bit rate for a given video image quality. This work is followed by [57], where variable segment lengths across different representations are considered—it is proposed that for higher bit rates, longer segments are used in order to improve coding efficiency.

Among the server-side actions is also the selection of compression algorithms for audiovisual content (in cases where it is not fixed by the system specification). A recent comparison of different video compression standards [58] justifies the very widespread use of H.264/AVC (Advanced Video Coding) encoding [59] for video as shown in Tables I and II, although codec flexibility, available in several proprietary and standard solutions, is a clear advantage due to the emerging highly efficient HEVC (High Efficiency Video Coding) standard [58].

Apart from single-layer codecs like H.264/AVC or HEVC where only different representations (i.e., different files) of the same video can be switched, multi-layer codecs can be used which enable bitstream switching. Such features were also introduced in AVC (cf. SP/SI synchronization/switching frames [60]) and date back to the MPEG-2 standard [61] which needed a big overhead to achieve scalability in times when processors were hardly fast enough.

A modern multi-layer codec is Scalable Video Coding (SVC) which is an amendment to AVC and offers temporal, spatial, and image quality scalability [62]. This means that SVC allows for adaptation of frame rate and content resolution, and switching between different levels of image quality. It makes use of difference coding of the video content such that data in lower layers can be used to predict data or samples of higher layers (so called enhancement layer). In order to switch to a higher layer, only the missing difference data have to be transmitted and added. Thus, the major difference to adaptation with single-layer codecs is that quality can be increased incrementally in case of spare resources. With single-layer codecs, on the other hand, a whole new segment has to be downloaded, and already downloaded lower quality segments have to be discarded.

There are two trade-offs that need to be addressed in case an SVC-based HAS solution is deployed. The first trade-off is the *overhead* introduced by multi-layer codecs. This means, for example, that an SVC file of a video of a certain quality is larger compared to an AVC file of the same video and quality. Fairly low overhead can be achieved in case the scalable encoder is carefully adjusted, and the number of enhancement layers is low—e.g., in [63] the authors achieve around 10% bit rate overhead in an optimized encoder when quality scalability is applied on low resolution sequences with 5 extractable bit rate levels. The authors also show that the overhead for spatial scalability largely depends on the spatial scalability ratio, i.e., the resolution ratio of the successive layers. SVC is shown to be very efficient for spatial scalability ratio of 1/2 but rather inefficient for scalability ratio of 3/4 where the overhead is between 40% and 100%. Performance of quality scalability of SVC has been analyzed in [64], where the results claim that the overhead needed for two enhancement layers is 10%–30%. These findings are confirmed by [65] and [66], where a detailed objective performance analysis of SVC in different layer setups is performed, leading to a recommendation of creating a separate SVC stream for each resolution, thus using only the quality scalability feature of SVC in practical HAS systems. It is also shown in [65], [66] that the performance varies largely across different SVC encoder implementations. The second trade-off is the amount of *signaling* required. The authors of [67] show that SVC can reduce the risk of stalling by always downloading the base layer and optionally downloading the enhancement layers when there is enough available throughput. Such improved flexibility comes at the cost of increased signaling traffic as several HTTP requests are needed per segment. This implies that AVC performs better under high latencies, while SVC adapts more easily to sudden and temporary bandwidth fluctuations when using a small receiver buffer.

By applying a hierarchical coding scheme, SVC allows for the selection of a suitable sub-bitstream for the on-the-fly adaptation of the media bitstream to device capabilities and current network conditions. A valid sub-bitstream contains at least the AVC-compatible base layer and zero or more enhancement layers. Note that all enhancement layers depend on the base layer and/or on the previous enhancement layer(s) of the same scalability dimension. The subjective evaluation of various SVC configurations is well known [68], [69] but papers dealing with DASH and QoE focus on AVC, and the integration of DASH and SVC is only evaluated using objective metrics [70] and simulations [71], [72] so far.

The architecture of HTTP adaptive streaming systems allows for optimization of the server load. In [73], for instance, the authors propose a scheme for balancing the load among several servers through altering the addresses in the DASH manifest files.

It is obvious that the main concern in the server-side actions is the selection of appropriate coding for the different available quality levels. This includes not only the selection of the compression algorithm and its settings, but also the decision on adaptation dimensions to be employed. An overview of the available adaptation dimensions will follow in Section V.

D. *Client-Side Actions*

On the client side, the most important decisions are which segments to download, when to start with the download, and how to manage the receiver video buffer. The adaptation algorithm (decision engine) should select the appropriate representations in order to maximize the QoE, which can be achieved in several different ways. The most common approach is to estimate the instantaneous channel bandwidth and use it as decision criterion.

The receiver video buffer size is dealt with in [74], where the authors perform an analysis of receiver buffer requirement for variable bit rate encoded bitstreams. They find that the optimal buffer length depends on the bitstream characteristics (data rate and its variance) as well as network characteristics and, of course, the desired video QoE represented by initial delay and rebuffering probability.

A recent work [75] reviews the available bitrate estimation algorithms and describes the active and passive bitrate measurement approaches. The passive measurement requires no additional probe packets to be inserted in the network, which results in no additional overhead at the expense of less accurate results. The absence of additional overhead is the reason why passive tools are generally used for available bitrate estimation in HAS. The author of [75] classifies the passive measurement approaches to *cross layer-based*, where the protocol stack is modified to obtain packet properties (e.g., [76]) and *model-based*, which employ throughput modeling (such as [77] for wireless LAN networks using TCP or UDP transport protocols). The following paragraphs describe different adaptation algorithms generally based on passive measurement of available bit rate or segment download time.

Depending on the actual use case and scenario, different adaptation strategies can be employed to adapt to the varying available bitrate. In [78] the authors compare several segment request strategies in HAS for live services. The analysis uses passive measurement of segment arrival times, aiming at evaluating the video startup delay, end to end delay, and the time available for segment download through the analysis of different initial delay adjustment, the time to start downloading the next segment, and the way to handle missing segments or their parts. It is shown that different strategies exhibit different behavior, and the adjustment needs to reflect network conditions and desired QoE priorities of the system.

For live and video on demand services, [79] and [80] describe a decision engine based on Markov Decision Process (MDP) using the estimated bit rate as input. Reference [81] proposes a rate adaptation algorithm based on smoothed bandwidth changes measured through segment fetch time, whereas in [82], the authors propose an adaptation engine based on the dynamics of the available throughput in the past and the actual buffer level to select the appropriate representation. At the same time, the algorithm adjusts the required buffer level to be kept in the next run. Also in [83], an algorithm for single-layer content (e.g., AVC) of constant bit rate is presented, which selects representations according to current bandwidth, current buffer level, and the average bit rate of each segment. For multi-layer content (e.g., SVC), [84] describes the Tribler algorithm, which relies on thresholds of downloaded segments, and [85] proposes the BIEB algorithm, which is also an SVC-based strategy computing segment thresholds based on size ratios between quality levels. Note that with multi-layer strategies different quality levels of the same time slot can be requested independently. Single-layer strategies can also request different representations of the same time slot, but only one can be used for decoding, and already downloaded other representations will be discarded.

It is important to mention that the presented algorithms select among the available representations just based on technical parameters like bandwidth or bit rate, but do not take the expected quality perceived by the end user into account.

E. Performance Studies

HTTP adaptive streaming uses the TCP transport protocol, which, although reliable, introduces higher overhead and delays compared to the simpler UDP, broadly used for video services earlier. As there are fundamental differences between TCP and UDP, studies have been done on justifying the use of TCP for video transmission. In [86], for instance, the authors employ discrete-time Markov models to describe the performance of TCP for live and stored media streaming without adaptation. It is found that TCP generally provides good streaming performance when the achievable TCP throughput is roughly twice the video bit rate, i.e., there is a significant system overhead as the expense for reliable transmission.

A number of studies have been published aiming at comparison of the existing HAS solutions, both proprietary and standardized, in terms of performance. In [83], the authors compare MSS, HLS, HDS, and DASH in a vehicular environment, using off-the-shelf client implementations for the proprietary systems and their own DASH client. They find that the best performance, represented by average achieved video bit rate and the number of switches among representations, is achieved by MSS among proprietary solutions and by Pipelined DASH among all the candidates. The idea behind Pipelined DASH is that several segments can be requested at a time in contrast to standard DASH. Pipelining is beneficial in vehicular and mobile scenarios, where packet loss might result in a poor usage of the available resources in case only one TCP connection is established. The drawback is that pipelining requires appropriate sending buffer control, i.e., server complexity is increased. More details on pipelining performance can be found in [87]. The authors of [88] compare the performance of MSS, Netflix, and OMSF (an open source player) clients in terms of the reaction of the clients to persistent or short-term bandwidth changes, the speed of convergence to the maximum sustainable bitrate and, finally, playback delay, important particularly for live content playback. The study reveals significant inefficiencies in each of the clients. In [89], the authors compare a MSS client and their own DASH client in Wireless Local Area Network (WLAN) environment, finding that the DASH client outperforms the MSS client in terms of average achieved bit rate, number of fluctuations, rebuffering time, and fairness. In [55], the performance of DASH for live streaming is studied. An analysis of performance with respect to segment size is provided, quantifying the impact of the HTTP protocol and segment size on the end to end delay.

The problem of the performance comparison in [83] and [88] is that the different clients are seen as closed components and the logic inside is unknown. In such cases, the system performance clearly depends on the actual implementation of the client and the adaptation algorithm used. In [85], a performance comparison of the adaptation algorithms described in [82]–[85] is conducted. The traffic patterns used for the evaluation were recorded in realistic wired and vehicular mobility situations. In terms of average playback quality and bandwidth utilization, BIEB [85] and Tribler [84] can outperform the other algorithms significantly. Both algorithms deliver a high average playback quality to the user, but Tribler has to switch to a different quality nine times more often than BIEB. The algorithm of [82] shows better results than BIEB in some aspects, as it has a lower quality switching frequency and a better network efficiency because no data is unnecessarily downloaded and bandwidth is wasted. However, compared to the size of the movie, the segments discarded by BIEB are negligible. In the vehicular scenario, BIEB outperforms the other algorithms, but no performance results are provided so far for other scenarios. Reference [90] enhances the performance comparison of [85] by computing the QoE-optimal adaptation strategy for each bandwidth condition and shows that the BIEB algorithm is closest to the optimum.

Also other optimization criteria for algorithm performance assessment like PSNR [91] or pseudo-subjective measures like engagement [92], [93] have been used. These criteria are often assumptions regarding QoE impact, which date from earlier studies and have neither been questioned nor verified with respect to their QoE appropriateness [94]. Thus, dedicated studies on the impact of adaptation strategies and application parameters on QoE will be presented in the following section. These results should be taken into account when designing a QoE-aware HAS algorithm.

IV. QoE Influence of Adaptation Strategy and Application Parameters

A QoE model for adaptive video streaming, which can be used for automated QoE evaluation, is described in [104]. The authors find that adaptation strategy related parameters (stalling, representation switches) have to be taken into account

TABLE III
EFFECTS OF APPLICATION PARAMETERS SETTINGS ON ADAPTATION AND ON SUBJECTIVELY PERCEIVED VIDEO QUALITY

No	**Adaptation** [29]	Yes
- More stalling - Worse bandwidth utilization		- Less stalling - Better bandwidth utilization
Small	**Buffer Size** [29]	Large
- More stalling - Less initial delay - Less memory requirements	Buffer size of 6 s is sufficient	- Less stalling - More initial delay - Increased memory requirements
Small	**Adaptation Interval (Segment Length)** [29]	Large
- Less stalling - More files - Lower coding efficiency - Worse quality - More switches - Shorter delay - Lower network utilization		- More stalling - Fewer files - Higher coding efficiency [54] - Better quality [95] - Less switches [95] - Higher delay [95] - Higher network utilization [96] - Negative impact on fairness [96]
Low	**Adaptation Frequency** [97]	High
- Better than constant low quality - Decreasing further has no effect - Time on individual layer becomes important [99], [100]	Switching is degradation itself [98]	- Annoying
Down	**Adaptation Direction** [98]	Up
- Stronger impact	QoE according to direction	- Smaller impact
Low	**Adaptation Amplitude** [97]	High
- Image quality change not detectable - Frame rate change down to 15 fps not detectable - Resolution change down to half of original size not detectable	Most dominant factor	- Low acceptance - Abrupt up-switching might increase QoE [101]
	Additional Effects	
	Recency effect [102]	- Higher quality in the end leads to higher QoE ([100]: only if less than 2 switches are present)
	Base layer [99], [102], [103]	- Higher base layer allows for longer impairments to be accepted

and that they have to be considered on a larger time scale (up to some minutes) than video encoding related parameters (resolution, frame rate, quantization parameter, bit rate), which only influence in the order of a few seconds. In [5], QoE metrics for adaptive streaming, which are defined in 3GPP DASH specification TS 26.247, are presented. They include HTTP request/response transactions, representation switch events, average throughput, initial delay, buffer level, play list, and MPD information. However, the conducted QoE evaluation considers only stalling as most dominating QoE impairment. Other results regarding the QoE influence of adaptation parameters are summarized in Table III and will be presented in more detail in this section.

Reference [29] investigates how playout buffer threshold and video segment size influence the number of stalling events. They find that a small buffer of 6 s is sufficient to achieve a near uninterrupted streaming experience under vehicular mobility. Further increasing the buffer size leads to an increased initial delay and could also be an issue for memory constrained mobile devices. With video segment size, there is a trade-off between short segment sizes resulting in many small files, which have to be stored for multiple bit rates of each video. Larger segment sizes, however, may not be sufficient to adapt to rapid bandwidth fluctuations especially in vehicular mobility and lead to more stalling. However, this effect can be balanced by increasing the buffer threshold, i.e., the amount of data which is buffered before the video playback starts. Thus, the authors state explicitly that it is important to configure the buffer threshold in accordance with the used video segment size. Also [95] confirms by simulations that a longer adaptation interval, i.e., longer time between two possible quality adaptations, leads to higher quality levels of the video and fewer quality changes. However, the number of stalling events and the total delay increase. Additionally, [96] reveals an impact on players' concurrent behavior, such that large segment sizes allow for a high network utilization but have negative effects on fairness. These aspects beyond pure video QoE will be discussed in more detail in Section VII.

Reference [98] shows that the active adaptation of the video bit rate improves or decreases the video quality according to the switching direction, but downgrading has a stronger impact on QoE than increasing the video bit rate. Thus, the authors argue that there could be a degradation caused by the switching itself. Reference [102] investigates the adaptation of image quality for layer-encoded videos. They find that the frequency of adaptation should be kept as small as possible. If a variation cannot be avoided, its amplitude should be kept as small as possible. Thus, a stepwise decrease of image quality is rated slightly better than one single decrease. Also [101] compares smooth to abrupt switching of image quality. They confirm that down-switching is generally considered annoying. Abrupt up-switching, however, might even increase QoE as users might be happy to notice the visual improvement. Reference [102] finds that a higher base (i.e., lowest quality) layer results in higher perceived quality, which implies that segments which raise the base layer should rather be downloaded instead of improving on higher quality layers. This finding is confirmed by [99] for mobile devices. Finally, [102] also observes a strong recency effect, i.e., higher quality in the end of a video clip leads to higher QoE. In [103], the impact of image quality adaptation on SVC videos is shown for a base layer and one enhancement layer. The authors find that a higher base layer quality allows for longer impairments to be accepted. The duration of such impairments has linear influence on the perceived quality. For their 12 s video clips the influence of the number of impairments is only significant between one and two impairments, while the interval between impairments does not seem to have any significant influence.

In [105], the authors present an approach which overrides client adaptation decisions in the network in order to optimize QoE globally or for a group of users. However, this way of "adaptation" goes beyond single user optimization as discussed within this section. A discussion on network level adaptation issues and related user experience problems follows in Section VII.

Reference [97] investigates flicker effects for SVC videos, i.e., rapid alternation of base layer and enhancement layer, in adaptive video streaming to handheld devices. They identify three effects, namely, the period effect, the amplitude effect, and the content effect. The period effect, i.e., the frequency of adaptation, manifests itself such that high frequencies (adaptation interval less than 1 s) are perceived as more annoying than constant low quality. At low frequencies (adaptation interval larger than 2 s), quality is better than constant low quality, but saturates when decreased further. The amplitude, i.e., the difference between quality levels, is the most dominant factor for the perception of flicker as artifacts become more apparent. However, image quality adaptation is not detectable for most participants at low amplitudes. Also for temporal adaptation, changes between frame rates of 15 fps and 30 fps are not detected by half of the users. Only an increase of the quantization parameter (QP), i.e., reduction of image quality, from 24 QP above 32 QP, or frame rate reduction below 10 fps brings significant flicker effects, which result in low acceptance for high frequencies. For spatial adaptation, the authors indicate that the change of resolution should not exceed half the original size in order to deliver a generally acceptable quality. Finally, the content plays a significant role in spatial and temporal adaptation. For image quality reduction, no significant effect can be found. The authors conclude that videos with complex spatial details are particularly affected by resolution reduction, while videos with complex and global motion require high frame rates for smooth playback.

In [100], the effects of switching amplitude, switching frequency, and recency effects are investigated. The results demonstrate the high impact of the switching amplitude, and that recency effects (i.e., impact of last quality level and time after last switch) can be neglected if more than two switches occur. Moreover, it turns out that not the switching frequency but rather the time on each individual layer has a significant impact on QoE. This confirms results from [99], which investigates dynamically varying video quality on mobile devices. In this work, the authors find that users reward attempts to improve quality, and thus, they suggest that quality should always be switched to a higher layer, if possible.

To sum up, adaptation is a key influence parameter of video streaming services. The reviewed studies suggest an influence of adaptation amplitudes and times (i.e., frequency, time on each layer), which has to be taken into account by HAS adaptation strategies. By setting the buffer and segment sizes, video service providers can adjust the adaptation times. As practically relevant switching frequencies (i.e., adaptation intervals of 2 s or more) are low and have little impact on QoE, adaptation algorithms should try to keep the quality as high as possible first. Additionally, the perceived quality is affected differently by the different adaptation dimensions (i.e., image quality, spatial, or temporal adaptation). Depending on the content, quality switches will be more or less perceivable. Thus, when preparing the streaming content, a content analysis could allow for improved video segmentation and selection of the best adaptation dimension(s) and amplitudes. Section V presents these dimensions in detail, and their influences on QoE is outlined in Section VI.

V. Video Adaptation Dimensions

In order to follow the requirement of providing video content at different bit rates for HTTP adaptive streaming, one or several adaptation dimensions can be utilized. In the following paragraphs, we describe the possible adaptation dimensions and provide a real world example of the bit rate reduction efficiency of each approach. Our real world example is based on encoding 20 seconds long sequences with different content (sport-200 m sprint, cartoon-a clip from the Sintel movie [106], action-a car chasing scene from the movie "Knight and Day"). We have encoded the sequences with H.264/AVC with varying frame rate (from 25 fps down to 2.5 fps), resolution (from 1920×1080 progressive down to 128×72 progressive), and quantization parameter (from 30 up to 51). All other encoding parameters remained unchanged during the experiment (the x264 codec implementation was used, high profile, level 4.0, adaptive GOP length up to 2 seconds).

Video Frame Rate based bit rate adaptation relies on decreasing the temporal resolution of a video sequence, i.e., encoding

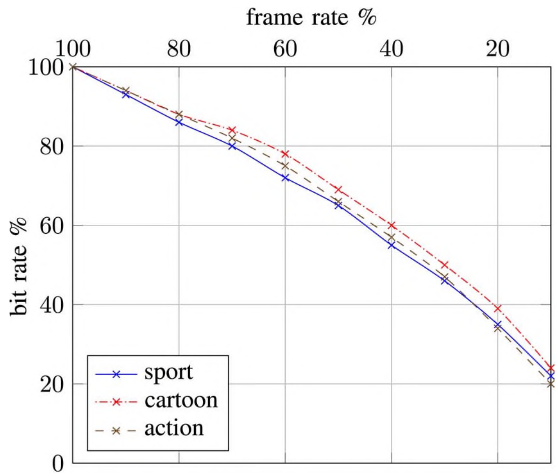

Fig. 7. Adaptation through frame rate reduction.

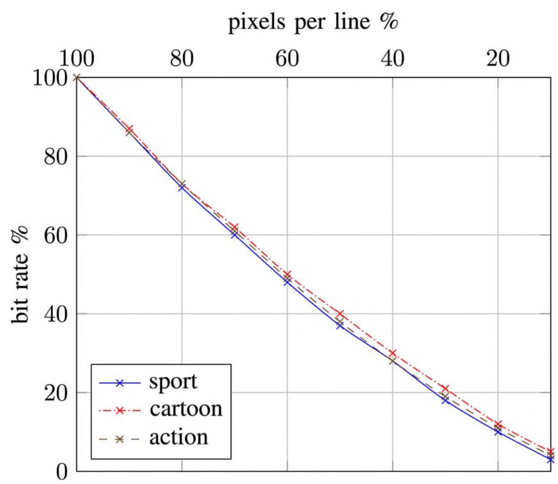

Fig. 8. Adaptation through resolution reduction.

a lower number of frames per second. The typical efficiency of such an approach is shown in Fig. 7. The original frame rate of a progressive-scanned video sequence corresponding to 100% is 25 fps in our real world example. In order to achieve 80% of the original bit rate, one needs to reduce the frame rate to approximately 65%. In such a case, motion in video is no longer perceived as smooth and the perceived quality degradation is significant [107].

Resolution based bit rate adaptation decreases the number of pixels in the horizontal and/or vertical dimension of each video frame. The corresponding efficiency of such an approach is shown in Fig. 8.[1] The steeper descend (compared to Fig. 7) of the curves is quite advantageous—even a small decrease of frame resolution leads to a significant reduction of required bit rate (e.g., 80% of the original bit rates is achieved by decreasing the frame size to approximately 85% in both directions).

Quantization based bit rate adaptation adjusts the lossy source encoder in order to reach the desired bit rate. In H.264/AVC, the available values of the quantization parameter (QP) are between 0 (lossless coding) and 51 (coarse quantiza-

[1]Resolution was changed in both the vertical and the horizontal dimension in order to keep the aspect ratio unchanged.

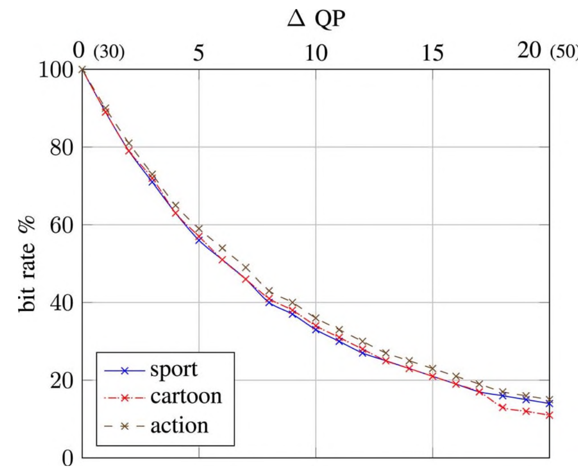

Fig. 9. Adaptation through adjustment of transform coefficient quantization.

tion with poor visual quality of a sequence). Fig. 9 shows the bit rate descend for quantization parameter increasing from 30 up to 50. The steep decrease of bit rate around QP 30 is getting flatter for QP values close to 50, which is quite natural as there is a certain amount of information in the encoded bitstream carrying data other than just quantized transform coefficients (e.g., prediction mode signalization, motion vector values, etc.).

It is obvious from Figs. 7–9 that the curves in each plot are very similar even for different content. This fact is a consequence of putting the relative bit rates on the ordinate instead of the absolute values, which vary greatly depending on the spatiotemporal properties of the different contents.

The adaptation dimensions mentioned in this section can be further extended as described in [108], where the author proposes a three-level model to describe the user satisfaction. Apart from transcoding, which is essentially the core operation for HAS content preparation, [108] also mentions transmoding, i.e., conversion among different modalities—audio, video, or text, as an alternative approach to adaptation. An example of a transmoding implementation can be found in [109].

VI. QoE Influence of Video Adaptation Dimensions

In this section, we review related work with respect to Quality of Experience of HTTP adaptive streaming. Therefore, we mainly focus on studies based on subjective user tests as these represent the gold standard for QoE assessment. Please note that most of these studies are not specifically designed for HAS, but the results are transferred where possible. However, some general issues still remain which are not addressed by research so far, e.g., long video QoE tests in the order of 10 minutes, which is a typical duration for user-generated content.

Several works (e.g., [118]–[120]) conclude that there is a content dependency regarding quantitative effects of adaptation. Especially spatial and temporal information of the video clips determine how the effects of adaptations are perceived. Thus, in this section no absolute results can be presented as they differ for each single video, but the main focus will be the presentation of general, qualitative effects of adaptation on QoE. Thereby, the different adaptation dimensions and their respective influence will be outlined first, and links to QoE

TABLE IV
MAIN QOE FINDINGS OF SINGLE DIMENSION ADAPTATION. MORE REFERENCES CAN BE FOUND IN THE DETAILED DESCRIPTION IN SECTION VI

Dimension	Main Findings	QoE Function
Image Quality	- Image quality decrease leads to lower QoE [110] - Encoder has most significant effect [112] - There is a logistic relationship between coded bit rate and QoE [113]	- Objective per-frame/per-chunk metrics [111]
Spatial	- Lower resolution leads to lower QoE [114] - Higher resolution can be a drawback with increased level of distortion [116]	- Resolution, temporal and spatial information [115]
Temporal	- Stalling is worst degradation [28] - Lower frame rate leads to lower QoE [117] - QoE of decreased frame rate is highly dependent on content motion [118]	- Number and length of stalling events [4] - Frame rate and spatial information [115]

models will be provided. A summary of this survey can be found in Table IV, but more details are given in the following three subsections. Afterwards, trade-offs between the different dimensions will be highlighted. Thus, this section will provide valuable guidelines when selecting the adaptation dimensions and preparing the content for a HAS streaming service.

A. Image Quality Adaptation

Reference [112] finds that the perceptual quality of a decoded video is significantly affected by the encoder type. They confirm work from 2005 in which the authors of [121] show that the video quality produced by the H.264 codec is the most satisfying, rated higher than RealVideo8 and H.263. Also for low bit rates H.264 outperforms H.263 and also MPEG-4 [122]. However, more advanced codecs like H.265/HEVC [123]–[125] and VP9 [126], [127] will become relevant for HAS in the next years.

The fact that the encoder type significantly affects QoE is also shown for scalable video codecs in [120], which investigates adaptation with H.264/SVC and wavelet-based scalable video coding. Reference [56] proposes an improved approach for encoding and segmentation of videos for adaptive streaming using H.264, which reduces the needed bit rate by up to 30% without any loss in quality. Conversely, this means that by using their approach, the video can be encoded with a higher image quality for the same target bit rate.

In [98], the performance of both H.264 and MPEG-4 is investigated under different mobile network conditions (WiFi and HSDPA) and video bit rates. Therefore, the authors implement their own on-the-fly video codec changeover and bit rate switching algorithm. With WiFi connection, H.264 is perceived better than MPEG-4 especially for low video bit rates. In contrast, with HSDPA connection, MPEG-4 yields better results for all bit rates. A change of codec during playback from H.264 to MPEG-4 always degrades the user perceived quality. However, a change in the other direction is perceived as an improvement for low bit rates. Furthermore, the authors find that a bit rate decrease, i.e., a decrease of image quality, results in a decreased quality, but a bit rate increase is not always better for QoE as the switch itself might be perceived as an impairment.

References [110], [113], and [128] investigate different video bit rates for different codecs and show that an increased video bit rate leads to an increased video quality. In particular in [113], a logistic function describes the relationship between coded bit rate and subjective video quality for H.264, stating that with increasing bit rate the video quality increases but eventually saturates. Thus, a further increase of bit rate does not result in a higher perceived quality.

Changing the quantization parameter of H.264 video streams is in the focus of [14]. The authors find that QoE falls slowly when the quantization parameter starts to increase, i.e., the video bit rate decreases and the image quality gets worse. Only after reaching a high quantization parameter the perceived quality drops faster. Reference [112] finds that in order to reach a good or excellent QoE the pixel bit rate should be around 0.1 bits per pixel when H.264 is used. If other information like frame rate or frame size are unavailable, pixel bit rate can serve as a rough quantitative gauge for QoE.

Reference [129] uses linear models and per-chunk metrics to predict the MOS of video sequences with image quality adaptations. Reference [111] shows that the perceived quality of HTTP video streams with dynamic quantization parameter adaptation can be predicted by temporal pooling of objective per-frame metrics. The authors state that a simple method like the mean of quality levels of a HAS video sequence delivers already very decent prediction performance.

The reviewed works suggest that for video streaming in general the video encoder has a significant effect on the perceived quality. Thus, the usage of H.264 is currently recommended also for HAS. For the highest quality layer, the QP can be adjusted such that good or excellent video quality is reached with minimal bit rate. When adapting the image quality, an increase of the QP leads to only slightly decreasing QoE in the beginning. Together with the convex shape of Fig. 9, it follows that HAS can utilize image quality adaptation by a small increase of QP in order to significantly reduce the bit rate while introducing a rather small degradation of QoE. However, high QPs lead to poor video quality and should be avoided.

B. Spatial Adaptation

References [114] and [130] find that spatial resolution is the key criterion for QoE for small screens. Low resolutions contribute to enhanced eyestrain of the subjects. However, acceptability of spatial resolutions is also tied to shot types (e.g., long shot, close-up). Reference [112] finds that, in general, higher MOS is associated with higher spatial resolution. However, they show that for the same video bit rate, a video with higher spatial resolution is perceived worse. This is due to a lower pixel bit rate value (cf. Section VI-A) which causes severe intra-frame degradations especially for sequences with large spatiotemporal activities. In [116], the impact of resolution on subjective quality is investigated by comparing high definition (HDTV) and standard definition television (SDTV) sequences. They find that QoE increases with increasing resolution for slightly distorted images. However, larger image size becomes a drawback when the level of distortion increases as artifacts are more prevalent and visible in HDTV. In this case, observers tend to prefer SD as this reduces the visual impact of the distortions.

In [115], a model for mapping resolution to MOS is presented. The considered resolutions range from SD to SQCIF. The authors show that MOS is a linear function of the logarithm of the resolution, spatial information, and temporal information. Moreover, they find that temporal information is more important than spatial information for their model.

As the display capability limits the displayed resolution, HAS should first adjust the delivered spatial resolution to the end user's device in order to avoid transmission of unnecessary data. A linear relation of resolution reduction and bit rate savings (see Fig. 8) can be observed. The impact of adaptation by resolution reduction depends mainly on the content and the device, but in general, resolution reduction leads to a lower image quality, and thus, to a lower QoE. However, in combination with image quality adaptation, spatial adaptation can even be beneficial for HAS. In particular, decreasing the image size when image quality is poor can help in order to obfuscate artifacts.

C. Temporal Adaptation

In [117], the effect of frame dropping on perceived video quality is investigated and a model for predicting QoE is proposed. It is shown that frame dropping has a negative impact on QoE and the quality impairment depends on motion and content of the sequence. Moreover, the authors find that periodic frame dropping, i.e., decrease of frame rate, is less annoying than an irregular discarding of frames. Note that stalling, i.e., a playout interruption by delaying or skipping the playout of several consecutive frames, can also be considered a temporal adaptation. However, the QoE impact of stalling is already discussed in Section II-B, and HAS primarily tries to avoid stalling. Therefore, this section focuses on the reduction of frame rate as a means of temporal quality adaptation, which has a less negative impact on QoE than stalling [28].

Reference [131] investigates the influence of frame rate on the acceptance of video clips of different temporal nature (e.g., still image, fast motion) and different importance of auditory and visual components (e.g., music video, sport highlights). The authors show that reducing the frame rate generally has a negative influence on the users' acceptance of the video clips. However, low temporal videos, i.e., videos with little motion, are affected more by lower frame rates than high temporal videos. This finding is confirmed by [118] which investigates different frame rates for dynamic content. They show that reducing the frame rate generally leads to a lower user satisfaction, but it does not proportionally reduce the users' understanding and perception of the video. Instead they find a complex link between understanding and perception, i.e., users have difficulty to absorb audio, video, and textual information concurrently. Thus, highly dynamic clips for which it is difficult to assimilate all information, such as sports or action clips, are unaffected by reduced frame rate. On the other hand, a static news clip, which can be understood easily as the most important information is delivered by audio, suffers from reduced frame rate because the lack of lip synchronization is clearly visible. Also [132] reconfirms the findings that a reduction of frame rate has only little impact on high motion videos. Reference [133] confirms these findings in a study on MPEG1 subjective quality which additionally takes into account packet loss, and states that the MOS is good if the frame rate is more than about 10 frames per second (fps). More recently, [134] states that the threshold of subjective acceptability is around 15 fps. Reference [135] finds that the optimal frame rate for a given bit rate depends on the type of motion in a sequence. They find that videos with jerky motion benefit from increased image quality at lower frame rates. Clips with smoother (i.e., less jerky) motion are generally insensitive to changes in frame rate.

Reference [28] finds that only for very low frame rates the quality decreases as the impairment duration increases. For medium or high frame rates the quality is similar whether the frame rate reduction occurs during the entire video or during a short part of the video. Thus, they state that there is no quality gain by re-increasing frame rate after a temporary drop.

In [115], a model for mapping frame rate to MOS is given. MOS can be expressed as a linear function of the logarithm of frame rate and spatial information. Adding temporal information does not improve the model performance.

The presented studies show that, in general, a lower frame rate leads to lower QoE. However, the impact of temporal adaptation depends heavily on the video content. Consequently, HAS can take advantage of frame rate adaptation especially for high motion content where small frame rate reductions are less visible. Drawbacks for HAS are that the frame rate cannot be reduced too much until the video quality is perceived as bad, and the bit rate savings in the relevant range are small compared to the other dimensions (cf. Fig. 7).

D. Trade-Offs Between Different Adaptation Dimensions

The three dimensions presented not only allow for single dimension adaptation but also for combined quality changes in multiple dimensions. Several studies consider trade-offs between different adaptations and are be presented in this section. Table V summarizes the main findings and links to the corresponding works.

Reference [94] claims that there exists an encoding which maximizes the user-perceived quality for a given target bit rate which can be extended to an optimal adaptation trajectory for a whole video stream. In their work, the authors focus on the adaptation of MPEG-4 video streams within a two-dimensional adaptation space defined by frame rate and spatial resolution. They show that a two-dimensional adaptation, which reduces both resolution and frame rate, outperforms adaptation in one dimension. Comparing clips of similar average bit rates, it is shown that reduction of frame rate is perceived worse than

TABLE V

TRADE-OFFS BETWEEN DIFFERENT ADAPTATION DIMENSIONS. A ▷ B MEANS THAT DIMENSION A IS MORE IMPORTANT THAN DIMENSION B, I.E., A DEGRADATION OF A IS WORSE THAN A DEGRADATION OF B. FOR EXAMPLE, STALLING WAS SHOWN TO BE A WORSE DEGRADATION THAN ADAPTATION (STALLING ▷ ADAPTATION). NOTE THAT SOME RESULTS FROM THE SURVEYED WORKS ARE CONTRADICTORY AND MIGHT DEPEND ON THE VIDEO TYPE. IN THESE CASES, INFORMATION ON THE VIDEO TYPES AND REFERENCES TO THE STUDIES CAN BE FOUND IN THE SECOND COLUMN. MORE DETAILS ON THE TRADE-OFFS ARE PRESENTED IN SECTION VI-D

Major Finding	Detailed Description
Stalling ▷ Adaptation [5]	- Stalling ▷ Initial Delay [112] - Stalling ▷ Image Quality [14] - Stalling ▷ Frame Rate [28]
Image Quality ▷ Frame Rate	- Spatially complex videos [119] - Slow motion videos [132] - Small screen devices [136] - Fast foreground motion [137] - High frame difference and variance [112]
Frame Rate ▷ Image Quality	- Fast camera or background motion [137] - High bit rates [120] - Low temporal activity [112]
Image Quality ▷ Resolution	- Different types of videos [132]
Frame Rate ▷ Resolution	- Different types of videos [94]
Resolution ▷ Frame Rate	- Low bit rates [120]

reduction of resolution. In [119], the authors research optimal combinations of image quality and frame rate for given bit rates. They find that until image quality improves to an acceptable level, it should be enhanced first. Once it is improved adequately, temporal quality should be improved. Especially spatially complex videos require a high image quality first, while videos with high camera motion require a higher frame rate at a lower bit rate. Also in [138], for a given bit rate trade-offs between frame rate and image quality are presented. A trend is found that for decreasing video bit rate also the optimal frame rate decreases. The authors show that for different video bit rates there exist switching points which define multiple bit rate regions requiring a different optimal frame rate for adaptation. Reference [136] investigates trade-offs between frame rate and quantization for soccer clips. The authors find that participants were more sensitive to reductions in frame quality than to reduced frame rate. Especially for small screen devices, a higher quantization parameter removes important information about the players and the ball. In contrast, a low frame rate of 6 fps is accepted 80% of the time although motion is not perceived as being smooth. The experimental results obtained by [139] show that image quality is valued higher by test users (which were able to choose which distortion step they preferred) than temporal resolution of the content for low bit rate videos. Reference [137] confirms that for fast foreground motion like soccer reducing frame rate is preferred to reducing frame quality. However, for fast camera or background motion a high frame rate is better because disturbing jerkiness can be detected more easily which results in lower QoE.

In [132], trade-offs between resolution and frame quality are investigated. They find that a small resolution (without upscaling) and high image quality is preferred to a large resolution and low frame quality for a given bit rate. Reference [120] compares different combinations of resolution, frame rate, and pixel bit rate, which result in similar average video bit rates. They find that at low bit rates a larger resolution is preferred, and thus, frame rate should be decreased. At high bit rates, frame rate is more important and pixel bit rate should be decreased to achieve a high perceived quality. Reference [112] maximizes the QoE by selecting an optimal combination of frame rate and frame size under limited bandwidth, i.e., video bit rate. They find that, in general, resolution should be kept low. For videos with a high frame difference and variance, also frame rate should be low (which implies a high pixel bit rate). Instead, frame rate should be high for content with low temporal activity in order to achieve a high QoE.

The major findings for HAS can be summarized as follows. First, an adaptation in multiple dimensions is perceived as better than a single dimension adaptation. Second, for most content types image quality is considered the most important dimension. Thus, reducing image quality too much will lead to bad QoE. This effect can be mitigated by reducing the image size at the same time to make artifacts less visible. Third, a high frame rate is important especially for content where jerkiness is easily visible (e.g., low motion content). Finally, resolution adaptation, which is closely related to image quality adaptation, is the least important dimension in most cases. Note that the impact of upscaling, which is a prevalent reaction of video players to reduced resolution and was inherent in the reviewed studies, depends on the specific player and could not be assessed in our survey.

VII. QoE DEGRADATIONS IN A SHARED NETWORK

Within this section we discuss challenges that arise from the interplay between HAS player instances and other applications that share the network. We thereby concentrate on effects that impair QoE of networked applications beyond video playback. This discussion includes several network related issues that arise from the particular behavior of video adaptation algorithms established in HAS clients, and their interplay with network level optimization algorithms. Within Section III-B, this interplay between application level control loop and network level control loop has already been introduced and depicted in Fig. 6. Typically such an interplay can introduce numerous different issues, especially as more network level control loops beyond TCP can be involved, e.g., in wireless networks. However, we not only address those issues that affect QoE relevant dimensions of HAS client instances, as discussed in previous sections, but also possible QoE impacts on other applications on the network. Our aim in this section is to identify the aforementioned issues and the resulting impairments on an application level, hence we do not go into protocol or network level details in terms of a detailed root cause analysis for these problems. Therefore, this section is very selective in terms of network issues and hence incomplete. A more thorough discussion of network related HAS problems can be found in [88] and [91]. The problems described here reach beyond pure video based quality, as they also tackle availability and stability of the single network connection the respective HAS client is utilizing, as well as the utilized network infrastructure at large. We first discuss issues between concurrent network entities (HAS clients, other networked applications), and second describe actionable measures that aim to counter these issues.

A. *Interactions Between Network Entities*

For the discussion of interactions between network entities and resulting issues, we distinguish two types of network entities: HAS client instances and other applications utilizing the same network. After this distinction, the following interactions are discussed: Between different HAS clients, other applications on HAS client instances, and HAS client instances on other applications. In addition, HAS client instances may also interfere with the TCP protocol as discussed at the end of this subsection.

1) Interactions Between HAS Players: If several adaptive players share a network connection, the following questions have been identified (cf. [88]) to be of particular interest:

- Can the players share the available bandwidth in a stable manner, without experiencing oscillatory bit rate transitions?
- Can they share the available bandwidth in a fair manner?
- Is bandwidth utilization influenced by competing HAS client instances?

One major issue for competing HAS players within a network is **stability** (in terms of quality levels) and the frequency of switching events. Several studies reveal that more frequent quality switching is invoked when more than one instance of HAS clients compete for bottleneck bandwidth [88], [140], [141]. Depending on the amount of available bandwidth and the time the different players join the stream the reaction is different. In [88], the authors show that in a two player setup the client joining the stream first grabs the highest available video bandwidth the network supports. Following, the second client joining starts with a low video bandwidth and tries to increase its video bandwidth subsequently. However, the throughput needed for the higher video bandwidth cannot be provided by the network as the first client utilizes this bandwidth already. Hence, the buffer level of the second client depletes, and the video adaptation algorithm switches to a lower video bit rate. This leads to a permanent oscillation between different video bit rates, and a large number of quality switching events for the second client, respectively. A similar finding is reported in [96] where a larger number of HAS client instances also results in increasing quality oscillations for all of these instances. The problem of the delay required to converge to the final bit rate is also associated to a high number of quality adaptation events. This impacts especially short viewing sessions like 2–3 min clips where the client will not reach the maximum available bit rate due to the above described behavior (cf. [140]).

Another challenge associated with the behavior of competing players is **fairness** between different HAS clients. If one player on network bottleneck grasps a large share of the bandwidth before the other players join the stream it will be privileged throughout its whole streaming session while the other players on the network compete for the remaining bandwidth as shown in [88], [140]. With a rising number of competing clients the unfairness increases [96]. The authors in [88] have shown that this behavior is not based on TCP's well known unfairness toward connections of different round trip times, but rather a result of the competing HAS client instances.

In addition to the abovementioned problems, **network utilization** within the bottleneck network is also impacted by competing HAS client instances. Results in [142] identify conservative strategies in the stream-switching logic as well as in the buffer controller as a major source of network underutilization for multiple client instances. This is in line with results described in [141] and [143] that report network underutilization as a result of video quality oscillation due to competing player instances.

Altogether, these three problems of stability, fairness, and bandwidth utilization do in turn influence QoE of HAS from an end user perspective. First, frequent quality oscillations due to instability have already been shown to degrade QoE (cf. Section IV). And second, low video quality levels as a result of unfair behavior or low network utilization do decrease resulting video QoE (cf. Section VI).

2) Interactions Between Other Applications and HAS Players: The impact of other applications on the quality of HAS clients has not been widely addressed yet. Investigations in this area discuss basic effects of parallel file downloads or browsing on the HAS control loop. Segments are downloaded sequentially with iterative decisions which segment quality to download next. Conservative decisions, as well as the additional delay introduced by the control loop may result in a reduced utilization of the network resources, and thus, a less than possible video quality [85]. This effect is amplified in case of parallel downloads. Depending on the specific implementation of the HAS control algorithm, particularly on its aggressiveness, less than the fair share of the resources is used [144]. Hence, parallel downloads account for more network resources resulting in a worse than necessary quality of the video stream.

Other applications with different traffic patterns like web browsing, gaming, or voice and video conferencing have not been investigated in this context. Hence, to foster a better understanding of the impact of multiple applications on the HAS control loop, additional research is required.

3) Interaction Between HAS Players and Other Applications: Beyond the above discussed influence of different HAS clients amongst each other, there is also the problem of their influence on other applications using the network connection. One of the main problems arising is the interaction between aggressive HAS clients, which periodically requests small files (video segments) over HTTP. This causes TCP to overestimate the bandwidth delay product of the transmission line and results in a buffer bloat effect as shown in [145] and [146], which in turn leads to queuing delays reaching up to one second and being over 500 ms for about 50% of the time. Having a one way queuing delay close to one second severely degrades QoE of cloud services [147], [148] and real-time communications services, such as VoIP [149] and video telephony [150]. Hence, it is almost impossible to use the bottleneck link for anything else but video transmission or large file downloads as shown in [146]. This is particularly concerning as [145] showed that active queue management (AQM) techniques, a widely believed solution to this problem, do not manage to eliminate large queuing delays.

4) Interactions Between HAS Players and TCP: The previous sections highlighted the interaction between HAS clients and between HAS clients and other applications. Besides that, the HAS control loop on application layer may also interfere with the TCP control loop resulting in performance issues without any other application. A TCP data transfer can be segmented into three phases [143]. In the initial burst phase, the congestion window is filled quickly resulting in aggressive probing to estimate the current congestion level. Once the window is full, the sender waits for acknowledgment packets. In the second phase, the ACK clocking phase, the congestion window increases slowly and most of the packets are transmitted upon receiving an ACK packet. TCP mechanisms like fast recovery and fast retransmission are fully working. In the third phase, the trailing ACK phase, the sender waits for the final ACKs, and fast retransmissions can not be used to notify the sender on corrupted or lost packets.

The performance of the data transmission is prone to packet losses in the first phase. Due to the aggressive packet probing, multiple packet losses may occur resulting in a delayed delivery of packets to the HAS client, and thus, a too conservative decision which quality to pick. In phase three, additional delays may be induced. Since fast retransmission may not be used, the packet timeout has to expire before a packet is retransmitted resulting in less throughput in this phase.

Due to the start-and-stop nature of HAS induced by the client-based control loop, more time is spent either in phase 1 or phase 3. Control delays may further result in an imprecise knowledge of the current congestion level in the network, a higher experienced packet loss, and thus, a lower throughput than theoretically possible.

B. Countermeasures

In the preceding paragraphs, problems due to entities competing for a bottleneck link have been identified. Within this section, we discuss how such problems can be counteracted in different network locations.

1) Server Based Approaches: On the server side, one can distinguish: a) Solutions related to properties of the adaptation set and related properties, such as segment sizes (cf. Section III-C), which can influence fairness and network utilization, and b) solutions that either actively select the chunk levels that are delivered to the player or interfere on TCP level. Within this section, we concentrate on the latter approaches and cross-reference to Section III-C for adaptation set related countermeasures.

In terms of active selection of chunk levels on the server side, [151] propose an algorithm that maximizes global perceptual quality in terms of video bitrate with respect to the bandwidth constraints for all clients on the network. To achieve that, they connect bitrate or quality level information with bandwidth availability and then decide which chunk levels are offered for a certain client instance. Thereby, they improve stability and network utilization. The approach proposed by [152] identifies quality oscillations based on the client requests. When oscillations are detected, the related server side algorithm limits the maximum chunk level available to the client in order to stabilize the stream quality level. Similar to the aforementioned approach, this results in improved stability and network utilization.

In contrast, the authors in [153] propose to place an upper bound on the TCP congestion window on the server side in order to control the burstiness of adaptive video traffic. As a result, packet losses and round trip time delays are reduced, which positively affects not only HAS but also other applications that share the network bottleneck.

Overall, the measures discussed within this section do tackle issues arising from interactions between different HAS instances as well as interactions between HAS instances and other applications.

2) Network Based Approaches: Purely network based approaches that overcome the previously mentioned limitations are sporadically discussed in literature. In [154], the authors propose a flexible redirection of HAS flows using Software-Defined Networking (SDN) to optimize the video playout quality. Reference [155] proposes to use Differentiated Services (DiffServ) to guarantee the video delivery. TCP rate shaping on a per flow level is introduced in [156] as one possibility to explicitly allocate resources to specific video flows, and thus, to enhance the QoE for the involved clients.

To conclude, the existing work mainly aims at optimizing the video quality for HAS clients and the investigation of interactions with other HAS clients and applications is not addressed in detail yet.

3) Proxy and Client Based Approaches: A purely client based solution is described in [157]. The proposed client solution does not request chunk levels (equivalent to a certain video bitrate) based on an available throughput estimation on the client side. Instead, the client buffer evolution over time is calculated and chunk levels are requested based on the relation of the current buffer level to a reference buffer level. Performance results show that stability and network utilization are increased compared to commercial HAS client implementations that utilize throughput estimation for determining chunk levels requested.

In contrast, the solution of [105] introduces a proxy server that monitors requested quality levels of all client flows passing through. This monitoring allows the proxy to identify the fairness distribution over the connected clients. In case clients with unfair shares of network throughput usage are identified, the proxy limits available chunk levels to the level of the other clients and thereby achieves fairness across connected clients.

A combination of client and proxy based solutions is proposed in [158] and [159]. The approach of [158] identifies each client's chunk level requests and calculates a median chunk level utilized in the network. This information is then distributed to the attached clients, which allows them to identify their own chunk level in comparison to other clients. Based on this, they adjust future requests such that a fair distribution of network resources is achieved. The solution described in [159] acts slightly different as the client buffer levels are shared with the proxy server, which then offers a certain range of chunk levels based on the according buffer levels. For both solutions, performance analysis shows that network utilization, stability, and fairness can be increased compared to networks containing standard HAS clients. Further improvements exploiting the interaction between a network-based proxy and HAS clients are presented in [160] and [161]. Both approaches aim at providing information to the HAS clients to select the appropriate video quality. In [161], available bandwidth measurements are forwarded to the clients, whereas in [160], the proxy defines which quality level shall be used. Thus, quality fluctuations can be reduced and the QoE between several HAS clients may be shared in a fair manner.

This section has shown that, beyond pure video QoE, the egoistic behavior of current adaptive video strategies results in interactions between two feedback loops (rate-adaptation logic at the application layer and TCP congestion control at the transport layer as depicted in Fig. 6). Unstable network conditions, an unfair distribution of network resources, and under-utilization of these resources are the result. These issues do not only impact HAS clients on the network but also severely impact other applications by large queuing delays and packet losses. Countermeasures addressing these issues exist and can be categorized into server based, network based, and proxy and client based approaches. However, each of these countermeasures tackles only a subset of the aforementioned issues, hence, research on a general solution to the problem is still needed.

VIII. Key Findings and Challenges From Different Stakeholder Perspectives

In this section, the lessons learned and best practices are summarized from the perspective of the different stakeholders involved in the HAS ecosystem. Throughout the survey, QoE was considered which reflects the end user perspective. In general, the end user is interested in optimal QoE for video streaming, but also an easy-to-use application, i.e., the user wants an app which just delivers good quality without need for manual configuration before or during service consumption. Further, additional end user aspects like energy consumption of smartphones or the used client bandwidth are of interest.

However, the end user is a pure consumer of the video service and cannot influence or interact with the service (although direct feedback might be considered in the future). The resulting video service quality depends on the stakeholders, which offer and deliver the service. Thus, we will consider the perspectives of these stakeholders, and present their goals and remaining challenges with respect to the QoE of HTTP adaptive video streaming:

A. Algorithm designer and programmer implementing the HAS solution in Section VIII-A
B. Network/Internet service provider in Section VIII-B
C. Video service/platform provider in Section VIII-C

A. HAS Algorithm Designer and Developer

Goals and Main Interests: The main goal of the HAS algorithm designer and developer is to provide optimal QoE for video streaming and to monitor the resulting QoE at the client side. Thereby, relevant QoE influence factors on different levels (system, content, user, context) are of interest. Client-side QoE monitoring allows direct feedback and the integration into the HAS algorithm in order to optimize the QoE of an individual user.

Lessons Learned:

- Stalling is the worst degradation and has to be avoided at costs of initial delay or quality adaptation.
- Buffer size of a few segment lengths is sufficiently large in practice to have less stalling.
- A holistic QoE model which describes the influences of each HAS parameter is missing.
- Not only technical parameters but also the expected quality perceived by the end user has to be taken into account for adaptation decisions.
- The most dominant adaptation factor is the adaptation amplitude.
- Algorithms should play out the highest possible quality.
- Adaptation frequency must not be too high to avoid flickering.
- Communication between clients and proxy may enhance user-perceived quality.
- Quality switches will be more or less perceivable depending on the concrete content, the motion pattern, and the selected adaptation dimension.

Challenges and Future Work: The adaptation algorithm (decision engine) should select the appropriate representations in order to maximize the QoE. The most important decisions are which segments to download, when to start the download, and how to manage the video buffer. If adaptation is necessary, the time of the quality switch should be based on the content in order to hide the switching or resulting degradation. Therefore, the algorithm designer requires a proper QoE model which considers the application-layer QoE parameters that can be influenced by the HAS algorithm. As different types of service are used (video on demand, live streaming) in different contexts, different algorithms (or parameter settings) have to be developed which need to be adjusted accordingly to the service requirements. Therefore, QoE and context monitoring has to be implemented and the monitored parameters need to be integrated in the algorithms' adaptation decision.

Some of the QoE influence factors can be measured directly at the client side, e.g., system level parameters related to application level (initial delay, stalling, representation switching) or related to device capabilities and screen resolution. Other influence factors like content level parameters (e.g., used video codec) or context level parameters (e.g., popularity) might be obtained directly from the video platform. Additionally, user preferences should be taken into account, as some users, for example, may prefer a very low resolution for news or other content instead of reduced frame rate. Including direct feedback into the adaptation decisions may overcome this problem, however, activation of users and possibly cheating/selfish behavior has to be tackled.

Current HAS algorithms are implemented in a network agnostic fashion (i.e., there is no direct information about the network conditions) and try to estimate/predict the network situation in the near future. Interfaces which allow an information exchange between application and network to specify service demands or to specify the network situation could be beneficial. The additional information from the network layer can be utilized by the algorithm to adjust the adaptation. Going one step beyond, whole cross-layer solutions (cf. Economic Traffic Management solutions for CDN services [162]) can be beneficial for all stakeholders and could be realized, for example, by Application-Layer Traffic Optimization (ALTO, [163]) or the northbound interface of Software-Defined Networking (SDN, [164]).

B. Network Provider Perspective

Goals and Main Interests: The network provider wants to efficiently utilize network resources and avoid unnecessary costs, e.g., inter-domain traffic or energy consumption of his network entities. Moreover, he wants to fulfill his SLAs and provide good QoS but also QoE to his customers for any services including HAS. Therefore, the network provider is interested in QoE monitoring (deployed in his network) to see if there are any problems in the network. Network dimensioning has to be applied in order to minimize resources and costs without degrading QoE. Additionally, the network provider cares about new business models and SLAs for high quality services (e.g., QoE tariffs) to attract new customers.

Lessons Learned:

- Problems stemming from the network can be identified, but there are many different options for influencing the data transport in the network.
- Other goal metrics (not only QoE) play an important role for the network provider (e.g., inter-domain traffic, SLAs).
- Video streaming performance is good when TCP throughput is roughly twice the video bit rate, i.e., there is a significant system overhead as the expense for reliable transmission.
- Egoistic client behavior can harm network wide HAS QoE and is best countered by employing either a centralized control unit or a cooperative client-proxy based control in order to establish fair QoE distribution across competing HAS instances

Challenges and Future Work: The network provider needs traffic and QoE models in order to understand the relevant QoE influence factors and the technical factors which can be influenced. QoE monitoring is required for network dimensioning, but also for traffic management to avoid QoE problems and resulting customer churn. Therefore, it is relevant to know which parameters to measure for QoE prediction or to foresee networking problems, and how to monitor those parameters technically (on which time scales, on which network elements, etc.). Existing measurement methodologies (e.g., [165] for YouTube QoE in 3G networks), the accuracies of each approach, and the costs/efforts for the implementation and operation (CAPEX/OPEX) are highly relevant for network providers, but would constitute an own survey.

The QoE monitoring results need to be aligned with concrete traffic management mechanisms (e.g., routing, caching) and service provisioning (e.g., dynamic bandwidth allocation, prioritization). Additionally, traffic fairness and net neutrality have to be taken into account, which is a non-trivial problem from a technical and law perspective. New traffic management approaches, which integrate information from QoE measurements but possibly also from other stakeholders, have to be developed to help network providers deliver a high service quality while reducing their expenses.

C. Video Service Provider Perspective

Goals and Main Interests: The main goals of the video service provider are optimal QoE and fairness for its end users. This will lead to a growing number of customers which will increase revenues. At the same time, the video service provider wants to minimize costs in terms of storage of videos, network bandwidth, and energy consumption for data centers and content delivery networks.

Lessons Learned:

- The usage of H.264 and its successors like H.265/HEVC is currently recommended also for HAS due to its efficiency.
- Multi-layer codecs allow more download flexibility since already downloaded parts of the video clip can be enhanced at a later time. This reduces the risk of stalling but requires increased signaling traffic as several HTTP requests are needed per segment.
- A H.264/SVC file of a video of a certain bit rate is larger compared to an H.264/AVC file of the same video (in same quality). AVC performs better under high latencies, while SVC adapts more easily to sudden and temporary bandwidth fluctuations when using a small receiver buffer.
- An adaptation in multiple dimensions is perceived as better than a single dimension adaptation.
- When preparing the streaming content, a content analysis could allow for improved video segmentation and selection of the best adaptation dimension(s).
- Larger segment sizes lead to improvements in QoE, higher coding efficiency, and higher network utilization. However, it has a negative impact on stalling and fairness.
- Image quality is more important for QoE than resolution due to compression artifacts.
- HAS can take advantage of frame rate adaptation, especially for high motion content where small frame rate reductions are less visible.
- Establishing a fair bandwidth distribution among clients and devices may increase the overall QoE of the clients.

- Reducing quality too much in any dimension will lead to bad QoE.

Challenges and Future Work: When preparing the content, the video service provider has to select appropriate parameters for segment length and representation bit rates. Moreover, the optimal encodings and adaptation dimensions have to be chosen. The offered representation levels and dimensions should be aligned to the adaptation algorithm on the end user side. A content analysis could be incorporated, such that possible quality switches can be hidden to the greatest extent.

Moreover, technical infrastructure is required to support the customers. This means, a powerful content delivery network is needed and proper mechanisms for optimal resource utilization and load balancing are required, which should be properly aligned with video codecs and video delivery protocols. The distribution of the video content has to place the content close to the end user in order to minimize transmission delays and increase the throughput. Therefore, smart content placement and caching strategies can be developed and integrated which utilize social information about the end users (e.g., TailGate [166], [167]). Additionally, fairness among customers should be taken into account, such that all users obtain the same service quality.

IX. Conclusion

In this work, the evolving research field of HAS was surveyed. To sum up, quality adaptation in video streaming and its influence on QoE is not well understood so far. It has been shown by related work that HAS clearly outperforms classical streaming as it significantly reduces stalling which is considered to be the worst quality degradation. As current solutions are not QoE-driven so far and only offer what can be called a "best effort QoE", this work outlined the influence of adaptation on QoE.

From investigating adaptation strategy parameters, it could be found that stalling, initial delay, memory requirements, and bandwidth utilization heavily depend on buffer size and segment size. However, a small buffer of a few segment lengths was shown to be sufficient for most bandwidth conditions. The most dominating factor is the adaptation amplitude, for which a high amplitude (i.e., a detectable quality change) results in low acceptance and perceived quality. Moreover, the adaptation frequency should be rather low, as switching is a degradation itself. Apart from these parameters, also time on each quality layer and base layer quality influence the QoE of users.

For each adaptation dimension, main findings and QoE functions have been presented. Multi-dimensional adaptation outperforms single dimension adaptation, and thus, should be considered in future HAS mechanisms and content preparation. The order of importance of the different adaptation dimension is image quality before frame rate and finally resolution, i.e., a decrease of image quality is perceived worst. Although this order seems to be valid for most video contents, there exist some video types for which the order can be different.

Beyond the impact of adaptation on pure video QoE, we also showed that QoE of other applications can be impaired due to network stability issues, high round trip times, and decreased TCP performance, which is caused by interactions between two feedback loops: HAS rate-adaptation logic at the application layer and TCP congestion control at the transport layer. In addition, we reviewed a number of measures that counter subsets of these interactions, ranging from server based approaches over network based approaches and collaborative solutions that utilize client-proxy communication.

We discussed numerous related works on the Quality of Experience of HAS in order to foster future research and development of new mechanisms. We showed that current HAS solutions only decide on adaptation based on bandwidth measurements and buffer levels. Hence, the resulting QoE, which is affected by adaptation, is not optimal. As a holistic model would be beneficial for all involved stakeholders, QoE researchers should aim at multidimensional QoE models, taking into account all facets of QoE and a systematic approach to measure HAS QoE. As the context of service consumption has a big influence on the perceived quality, a concept for context monitoring has to be developed and implemented. Moreover, collaborative solutions which include information from other stakeholders or direct feedback from end users have to be investigated on the technical side. The information obtained from other stakeholders can be beneficial in situations in which only limited information about system, content, user, or context is available and corresponding parameters are estimated. Thus, future HAS solutions should be QoE-driven, context aware, and collaborative, such that especially end users but also all other involved stakeholders benefit from improved adaptation decisions and improved quality of video streaming services.

Acknowledgment

The authors would like to thank the anonymous reviewers for their valuable and constructive comments and suggestions, as well as Stanislav Lange for proof reading. The authors alone are responsible for the content.

References

[1] "Cisco visual networking index: Forecast and methodology, 2012–2017," San Jose, CA, USA, Tech. Rep., 2013.
[2] J. Roettgers, "Don't touch that dial: How YouTube is bringing adaptive streaming to mobile, TVs," 2013. [Online]. Available: http://gigaom.com/2013/03/13/youtube-adaptive-streaming-mobile-tv
[3] A. Sackl, P. Zwickl, and P. Reichl, "The trouble with choice: An empirical study to investigate the influence of charging strategies and content selection on QoE," in *Proc. 9th Int. CNSM*, Zurich, Switzerland, 2013, pp. 298–303.
[4] T. Hoßfeld *et al.*, "Quantification of YouTube QoE via crowdsourcing," in *Proc. IEEE Int. Symp. Multimedia*, Dana Point, CA, USA, 2011, pp. 494–499.
[5] O. Oyman and S. Singh, "Quality of experience for HTTP adaptive streaming services," *IEEE Commun. Mag.*, vol. 50, no. 4, pp. 20–27, Apr. 2012.
[6] Y. Wang, "Survey of objective video quality measurements," EMC Corp., Hopkinton, MA, USA, Tech. Rep. WPI-CS-TR-06-02, 2006.
[7] U. Engelke and H.-J. Zepernick, "Perceptual-based quality metrics for image and video services: A survey," in *Proc. 3rd EuroNGI Conf. Netw.*, Trondheim, Norway, 2007, pp. 190–197.
[8] W. Lin and C.-C. J. Kuo, "Perceptual visual quality metrics: A survey," *J. Vis. Commun. Image Represent.*, vol. 22, no. 4, pp. 297–312, May 2011.
[9] S. Chikkerur, V. Sundaram, M. Reisslein, and L. Karam, "Objective video quality assessment methods: A classification, review, performance comparison," *IEEE Trans. Broadcast.*, vol. 57, no. 2, pp. 165–182, Jun. 2011.

[10] N. Zong, Survey and gap analysis for HTTP streaming standards and implementations, Internet Engineering Task Force Network Working Group, Fremont, CA, USA, Tech. Rep. [Online]. Available: http://tools.ietf.org/html/draft-zong-httpstreaming-gap-analysis-01

[11] K. J. Ma, R. Barto, and S. Bhatia, "A survey of schemes for Internet-based video delivery," *J. Netw. Comput. Appl.*, vol. 34, no. 5, pp. 1572–1586, Sep. 2011.

[12] V. Adzic, H. Kalva, and B. Furht, "A survey of multimedia content adaptation for mobile devices," *Multimedia Tools Appl.*, vol. 51, no. 1, pp. 379–396, Jan. 2011.

[13] H. Luo and M.-L. Shyu, "Quality of service provision in mobile multimedia—A survey," *Human-Centric Comput. Inf. Sci.*, vol. 1, no. 5, pp. 1–15, Nov. 2011.

[14] K. D. Singh, Y. Hadjadj-Aoul, and G. Rubino, "Quality of experience estimation for adaptive HTTP/TCP video streaming using H.264/AVC," in *Proc. IEEE CCNC*, Las Vegas, NV, USA, 2012, pp. 127–131.

[15] "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.

[16] R. K. P. Mok, E. W. W. Chan, X. Luo, and R. K. C. Chan, "Inferring the QoE of HTTP video streaming from user-viewing activities," in *Proc. ACM SIGCOMM W-MUST*, Toronto, ON, Canada, 2011, pp. 31–36.

[17] T. Hoßfeld *et al.*, "Initial delay vs. interruptions: Between the devil and the deep blue sea," in *Proc. 4th Int. Workshop QoMEX*, Yarra Valley, Vic., Australia, 2012, pp. 1–6.

[18] S. Egger, P. Reichl, T. Hoßfeld, and R. Schatz, "Time is bandwidth? Narrowing the gap between subjective time perception and quality of experience," in *Proc. IEEE ICC*, Ottawa, ON, Canada, 2012, pp. 1325–1330.

[19] R. E. Kooij, A. Kamal, and K. Brunnström, "Perceived quality of channel zapping," in *Proc. CSN*, Palma de Mallorca, Spain, 2006, pp. 155–158.

[20] S. Egger, T. Hoßfeld, R. Schatz, and M. Fiedler, "Waiting times in quality of experience for web based services," in *Proc. 4th Int. Workshop QoMEX*, Yarra Valley, Vic., Australia, 2012, pp. 86–96.

[21] A. Sackl, S. Egger, and R. Schatz, "Where's the music? Comparing the QoE impact of temporal impairments between music and video streaming," in *Proc. 5th Int. Workshop QoMEX*, Klagenfurt, Austria, 2013, pp. 64–69.

[22] A. Raake and S. Egger, "Quality and quality of experience," in *Quality of Experience: Advanced Concepts, Applications and Methods*, S. Möller and A. Raake, Eds. New York, NY, USA: Springer-Verlag, 2014.

[23] T. De Pessemier, K. De Moor, W. Joseph, L. De Marez, and L. Martens, "Quantifying the influence of rebuffering interruptions on the user's quality of experience during mobile video watching," *IEEE Trans. Broadcast.*, vol. 59, no. 1, pp. 47–61, Mar. 2013.

[24] A. Finamore, M. Mellia, M. M. Munafò, R. Torres, and S. G. Rao, "YouTube everywhere: Impact of device and infrastructure synergies on user experience," in *Proc. Internet Meas. Conf.*, Berlin, Germany, 2011, pp. 345–360.

[25] L. Chen, Y. Zhou, and D. M. Chiu, "Video browsing—A study of user behavior in online VoD services," in *Proc. 22nd ICCCN*, Nassau, Bahamas, 2013, pp. 1–7.

[26] Y. Qi and M. Dai, "The effect of frame freezing and frame skipping on video quality," in *Proc. 2nd Int. Conf. IIH-MSP*, Pasadena, CA, USA, 2006, pp. 423–426.

[27] T. N. Minhas and M. Fiedler, "Impact of disturbance locations on video quality of experience," in *Proc. 2nd Workshop QoEMCS*, Lisbon, Portugal, 2011, pp. 1–5.

[28] Q. Huynh-Thu and M. Ghanbari, "Temporal aspect of perceived quality in mobile video broadcasting," *IEEE Trans. Broadcast.*, vol. 54, no. 3, pp. 641–651, Sep. 2008.

[29] J. Yao, S. S. Kanhere, I. Hossain, and M. Hassan, "Empirical evaluation of HTTP adaptive streaming under vehicular mobility," in *Proc. 10th Int. IFIP TC 6 Netw. Conf.*, Valencia, Spain, 2011, pp. 92–105.

[30] T. Zinner, T. Hoßfeld, T. N. Minash, and M. Fiedler, "Controlled vs. uncontrolled degradations of QoE—The provisioning–delivery hysteresis in case of video," in *Proc. 1st Workshop QoEMCS*, Tampere, Finland, 2010, pp. 1–3.

[31] Move Networks, Move Networks 2010. [Online]. Available: http://www.movenetworkshd.com

[32] A. Zambelli, "A history of media streaming and the future of connected TV," *The Guardian*, 2013. [Online]. Available: http://www.theguardian.com/media-network/media-network-blog/2013/mar/01/history-streaming-future-connected-tv

[33] D. F. Brueck and M. B. Hurst, "Apparatus, system, method for multi-bitrate content streaming," U.S. Patent 7 818 444 B2, Oct. 19, 2010.

[34] A. Zambelli, Smooth streaming technical overview, Microsoft Corp., Redmond, WA, USA, Tech. Rep. [Online]. Available: http://www.iis.net/learn/media/on-demand-smooth-streaming/smooth-streaming-technical-overview

[35] Apple Inc., HTTP Live Streaming Overview 2013. [Online]. Available: https://developer.apple.com/library/ios/documentation/networkinginternet/conceptual/streamingmediaguide/Introduction/Introduction.html

[36] Adobe Systems Inc., HTTP Dynamic Streaming 2013. [Online]. Available: http://www.adobe.com/products/hds-dynamic-streaming.html

[37] Adobe Systems Inc., HTTP Dynamic Streaming on the Adobe Flash Platform 2010. [Online]. Available: https://bugbase.adobe.com/index.cfm?event=file.view&id=2943064&seqNum=6&name=httpdynamicstreaming_wp_ue.pdf

[38] European Telecommunications Standard Institute (ETSI). (2009). Universal Mobile Telecommunication System (UMTS); LTE; Transparent end-to-end Packet-Switched Streaming Service (PSS); Protocols and Codecs, Sophia-Antipolis Cedex, France, 3GPP TS 26.234 Version 9.1.0 Release 9.

[39] European Telecommunications Standard Institute (ETSI). (2010). Universal Mobile Telecommunication System (UMTS); LTE; Transparent end-to-end Packet-Switched Streaming Service (PSS); Progressive download and dynamic adaptive streaming over HTTP (3GP-DASH), Sophia-Antipolis Cedex, France, 3GPP TS 26.247 Version 1.0.0 Release 10.

[40] *Information Technology—Dynamic Adaptive Streaming Over HTTP (DASH)—Part 1: Media Presentation Description and Segment Formats*, ISO/IEC 23009-1:2012, 2012.

[41] DASH Industry Forum, For Promotion of MPEG-DASH 2013. [Online]. Available: http://dashif.org

[42] *Guidelines for Implementation: DASH-AVC/264 Interoperability Points*, DASH Industry Forum, 2013. [Online]. Available: http://dashif.org/w/2013/06/DASH-AVC-264-base-v1.03.pdf

[43] Wikipedia, 615782083 List of Codecs 2014. [Online]. Available: http://en.wikipedia.org/w/index.php?title=List_of_codecs&oldid=615782083, 615782083

[44] *Extensible Markup Language (XML) 1.0 (Fifth Edition)*, World Wide Web Consortium (W3C), Cambridge, MA, USA, 2013.

[45] *Flash Media Manifest (F4M) Format Specification*, Adobe Systems Inc., San Jose, CA, USA, 2013.

[46] M. Levkov, "Video Encoding and Transcoding Recommendations for HTTP Dynamic Streaming in the Adobe Flash Platform," 2010. [Online]. Available: http://download.macromedia.com/flashmediaserver/http_encoding_recommendations.pdf

[47] *HbbTV Specification*, HbbTV Association, Erlangen, Germany, 2012.

[48] *Information Technology—High Efficiency Coding and Media Delivery in Heterogeneous Environments—Part 2: High Efficiency Video Coding*, ISO/IEC 23008-2:2013, 2013.

[49] *Information Technology—Generic Coding of Moving Pictures and Associated Audio Information: Systems*, ISO/IEC 13818-1:2000, 2000.

[50] *Information Technology—Coding of Audio-visual Objects—Part 12: ISO Base Media File Format*, ISO/IEC 14496-12:2005, 2005.

[51] T. Siglin, Unifying global video strategies: MP4 file fragmentation for broadcast, mobile and web delivery, Transitions, Inc., Bellevue, KY, USA. [Online]. Available: http://184.168.176.117/reports-public/Adobe/20111116-fMP4-Adobe-Microsoft.pdf

[52] *F4V/FLV technology center*, Adobe Systems Inc., San Jose, CA, USA, 2013.

[53] I. Kofler, R. Kuschnig, and H. Hellwagner, "Implications of the ISO base media file format on adaptive HTTP streaming of H.264/SVC," in *Proc. IEEE CCNC*, Las Vegas, NV, USA, 2012, pp. 549–553.

[54] K. Ramkishor, T. S. Raghu, K. Siuman, and P. S. S. B. K. Gupta, "Adaptation of video encoders for improvement in quality," in *Proc. IEEE ISCAS*, Bangkok, Thailand, 2003, pp. II-692–II-695.

[55] T. Lohmar, T. Einarsson, P. Frojdh, F. Gabin, and M. Kampmann, "Dynamic adaptive HTTP streaming of live content," in *Proc. IEEE Int. Symp. WoWMoM Netw.*, Lucca, Italy, 2011, pp. 1–8.

[56] V. Adzic, H. Kalva, and B. Furht, "Optimizing video encoding for adaptive streaming over HTTP," *IEEE Trans. Consum. Electron.*, vol. 58, no. 2, pp. 397–403, May 2012.

[57] J. Lievens, S. Satti, N. Deligiannis, P. Schelkens, and A. Munteanu, "Optimized segmentation of H.264/AVC video for HTTP adaptive streaming," in *Proc. IFIP/IEEE Int. Symp. IM*, Ghent, Belgium, 2013, pp. 1312–1317.

[58] J. Ohm, G. Sullivan, H. Schwarz, T. K. Tan, and T. Wiegand, "Comparison of the coding efficiency of video coding standards—Including High Efficiency Video Coding (HEVC)," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 22, no. 12, pp. 1669–1684, Dec. 2012.

[59] *Information Technology—Coding of Audio-Visual Objects—Part 10: Advanced Video Coding*, ISO/IEC 14496-10:2012, 2013.
[60] M. Karczewicz and R. Kurceren, “The SP- and SI-frames design for H.264/AVC,” *IEEE Trans. Circuits Syst. Video Technol.*, vol. 13, no. 7, pp. 637–644, Jul. 2003.
[61] *Information Technology—Generic Coding of Moving Pictures and Associated Audio Information: Video*, ISO/IEC 13818-2:1996, 1996.
[62] H. Schwarz, D. Marpe, and T. Wiegand, “Overview of the scalable video coding extension of the H.264/AVC standard,” *IEEE Trans. Circuits Syst. Video Technol.*, vol. 17, no. 9, pp. 1103–1120, Sep. 2007.
[63] M. Wien, H. Schwarz, and T. Oelbaum, “Performance analysis of SVC,” *IEEE Trans. Circuits Syst. Video Technol.*, vol. 17, no. 9, pp. 1194–1203, Sep. 2007.
[64] R. Gupta, A. Pulipaka, P. Seeling, L. Karam, and M. Reisslein, “H.264 Coarse Grain Scalable (CGS) and Medium Grain Scalable (MGS) encoded video: A trace based traffic and quality evaluation,” *IEEE Trans. Broadcast.*, vol. 58, no. 3, pp. 428–439, Sep. 2012.
[65] M. Grafl *et al.*, “Scalable video coding guidelines and performance evaluations for adaptive media delivery of high definition content,” in *Proc. IEEE ISCC*, Split, Croatia, 2013, pp. 000855–000861.
[66] M. Grafl, C. Timmerer, H. Hellwagner, W. Cherif, and A. Ksentini, “Evaluation of hybrid scalable video coding for HTTP-based adaptive media streaming with high-definition content,” in *Proc. 14th Int. Symp. Workshops WoWMoM Netw.*, Madrid, Spain, 2013, pp. 1–7.
[67] J. Famaey *et al.*, “On the merits of SVC-based HTTP adaptive streaming,” in *Proc. IFIP/IEEE Int. Symp. IM*, Ghent, Belgium, 2013, pp. 419–426.
[68] J.-S. Lee *et al.*, “Subjective evaluation of scalable video coding for content distribution,” in *Proc. 18th ACM Int. Conf. Multimedia*, Florence, Italy, 2010, pp. 65–72.
[69] T. Oelbaum, H. Schwarz, M. Wien, and T. Wiegand, “Subjective performance evaluation of the SVC extension of H.264/AVC,” in *Proc. 15th IEEE ICIP*, San Diego, CA, USA, 2008, pp. 2772–2775.
[70] C. Müller, D. Renzi, S. Lederer, S. Battista, and C. Timmerer, “Using scalable video coding for dynamic adaptive streaming over HTTP in mobile environments,” in *Proc. 20th EUSIPCO*, Bucharest, Romania, 2012, pp. 2208–2212.
[71] R. Huysegems, B. De Vleeschauwer, T. Wu, and W. Van Leekwijck, “SVC-based HTTP adaptive streaming,” *Bell Labs Tech. J.*, vol. 16, no. 4, pp. 25–41, Mar. 2012.
[72] Y. Sanchez *et al.*, “Efficient HTTP-based streaming using scalable video coding,” *Signal Process., Image Commun.*, vol. 27, no. 4, pp. 329–342, Apr. 2012.
[73] Z. Li *et al.*, “Network friendly video distribution,” in *Proc. 3rd Int. Conf. NOF*, Gammarth, Tunisia, 2012, pp. 1–8.
[74] T. Kim, N. Avadhanam, and S. Subramanian, “Dimensioning receiver buffer requirement for unidirectional VBR video streaming over TCP,” in *Proc. IEEE ICIP*, Atlanta, GA, USA, 2006, pp. 3061–3064.
[75] T. Arsan, “Review of bandwidth estimation tools and application to bandwidth adaptive video streaming,” in *Proc. 9th Int. Conf. HONET*, Istanbul, Turkey, 2012, pp. 152–156.
[76] Z. Yuan, H. Venkataraman, and G.-M. Muntean, “iBE: A novel bandwidth estimation algorithm for multimedia services over IEEE 802.11 wireless networks,” in *Proc. 12th IFIP/IEEE Int. Conf. MMNS*, Venice, Italy, 2009, pp. 69–80.
[77] Z. Yuan, H. Venkataraman, and G.-M. Muntean, “MBE: Model-based available bandwidth estimation for IEEE 802.11 data communications,” *IEEE Trans. Veh. Technol.*, vol. 61, no. 5, pp. 2158–2171, Jun. 2012.
[78] T. Kupka, P. Halvorsen, and C. Griwodz, “An evaluation of live adaptive HTTP segment streaming request strategies,” in *Proc. IEEE 36th Conf. LCN*, Bonn, Germany, 2011, pp. 604–612.
[79] C. C. Wüst and W. F. J. Verhaegh, “Quality control for scalable media processing applications,” *J. Sched.*, vol. 7, no. 2, pp. 105–117, Mar. 2004.
[80] D. Jarnikov and T. Ozcelebi, “Client intelligence for adaptive streaming solutions,” in *Proc. IEEE ICME*, Singapore, 2010, pp. 1499–1504.
[81] C. Liu, I. Bouazizi, and M. Gabbouj, “Rate adaptation for adaptive HTTP streaming,” in *Proc. 2nd Annu. ACM Conf. MMSys*, Santa Clara, CA, USA, 2011, pp. 169–174.
[82] K. Miller, E. Quacchio, G. Gennari, and A. Wolisz, “Adaptation algorithm for adaptive streaming over HTTP,” in *Proc. 19th Int. PV Workshop*, Munich, Germany, 2012, pp. 173–178.
[83] C. Müller, S. Lederer, and C. Timmerer, “An evaluation of dynamic adaptive streaming over HTTP in vehicular environments,” in *Proc. 4th Workshop MoVID*, Chapel Hill, NC, USA, 2012, pp. 37–42.
[84] S. Oechsner, T. Zinner, J. Prokopetz, and T. Hoßfeld, “Supporting scalable video codecs in a P2P video-on-demand streaming system,” in *Proc. 21st ITC SS Multimedia Appl.—Traffic, Perform. QoE*, Miyazaki, Japan, 2010, pp. 48–53.
[85] C. Sieber, T. Hoßfeld, T. Zinner, P. Tran-Gia, and C. Timmerer, “Implementation and user-centric comparison of a novel adaptation logic for DASH with SVC,” in *Proc. IFIP/IEEE Int. Workshop QCMan*, Ghent, Belgium, 2013, pp. 1318–1323.
[86] B. Wang, J. Kurose, P. Shenoy, and D. Towsley, “Multimedia streaming via TCP: An analytic performance study,” *ACM Trans. Multimedia Comput., Commun. Appl.*, vol. 4, no. 2, pp. 16:1–16:22, May 2008.
[87] H. F. Nielsen *et al.*, “Network performance effects of HTTP/1.1, CSS1, PNG,” *ACM SIGCOMM Comput. Commun. Rev.*, vol. 27, no. 4, pp. 155–166, Oct. 1997.
[88] S. Akhshabi, A. C. Begen, and C. Dovrolis, “An experimental evaluation of rate-adaptation algorithms in adaptive streaming over HTTP,” in *Proc. 2nd Annu. ACM Conf. MMSys*, Santa Clara, CA, USA, 2011, pp. 157–168.
[89] K. Miller, N. Corda, S. Argyropoulos, A. Raake, and A. Wolisz, “Optimal adaptation trajectories for block-request adaptive video streaming,” in *Proc. 20th Int. PV Workshop*, San Jose, CA, USA, 2013, pp. 1–8.
[90] T. Hoßfeld, M. Seufert, C. Sieber, T. Zinner, and P. Tran-Gia, “Close to optimum? User-centric evaluation of adaptation logics for HTTP adaptive streaming,” PIK—Praxis der Informationverarbeitung und-Kommunikation (PIK) 2014.
[91] P. Pahalawatta, R. Berry, T. Pappas, and A. Katsaggelos, “Content-aware resource allocation and packet scheduling for video transmission over wireless networks,” *IEEE J. Sel. Areas Commun.*, vol. 25, no. 4, pp. 749–759, May 2007.
[92] F. Dobrian *et al.*, “Understanding the impact of video quality on user engagement,” in *Proc. ACM SIGCOMM*, Toronto, ON, Canada, 2011, pp. 362–373.
[93] A. Balachandran *et al.*, “A quest for an Internet video quality-of-experience metric,” in *Proc. 11th ACM Workshop HotNets-XI*, Redmond, WA, USA, 2012, pp. 97–102.
[94] N. Cranley, P. Perry, and L. Murphy, “User perception of adapting video quality,” *Int. J. Human-Comput. Stud.*, vol. 64, no. 8, pp. 637–647, Aug. 2006.
[95] O. Abboud, T. Zinner, K. Pussep, S. Al-Sabea, and R. Steinmetz, “On the impact of quality adaptation in SVC-based P2P video-on-demand systems,” in *Proc. 2nd Annu. ACM Conf. MMSys*, Santa Clara, CA, USA, 2011, pp. 223–232.
[96] R. Kuschnig, I. Kofler, and H. Hellwagner, “An evaluation of TCP-based rate-control algorithms for adaptive Internet streaming of H.264/SVC,” in *Proc. 1st Annu. ACM Conf. MMSys*, Phoenix, AZ, USA, 2010, pp. 157–168.
[97] P. Ni, R. Eg, A. Eichhorn, C. Griwodz, and P. Halvorsen, “Flicker effects in adaptive video streaming to handheld devices,” in *Proc. 19th ACM Int. Conf. MM*, Scottsdale, AZ, USA, 2011, pp. 463–472.
[98] B. Lewcio, B. Belmudez, A. Mehmood, M. Wältermann, and S. Möller, “Video quality in next generation mobile networks-perception of time-varying transmission,” in *Proc. IEEE Int. Workshop Tech. CQR*, Naples, FL, USA, 2011, pp. 1–6.
[99] A. K. Moorthy, L. K. Choi, A. C. Bovik, and G. De Veciana, “Video quality assessment on mobile devices: Subjective, behavioral and objective studies,” *IEEE J. Sel. Topics Signal Process.*, vol. 6, no. 6, pp. 652–671, Oct. 2012.
[100] T. Hoßfeld, M. Seufert, C. Sieber, and T. Zinner, “Assessing effect sizes of influence factors towards a QoE model for HTTP adaptive streaming,” in *Proc. 6th Int. Workshop QoMEX*, Singapore, 2014.
[101] M. Grafl and C. Timmerer, “Representation switch smoothing for adaptive HTTP streaming,” in *Proc. 4th Int. Workshop PQS*, Vienna, Austria, 2013, pp. 178–183.
[102] M. Zink, J. Schmitt, and R. Steinmetz, “Layer-encoded video in scalable adaptive streaming,” *IEEE Trans. Multimedia*, vol. 7, no. 1, pp. 75–84, Feb. 2005.
[103] Y. Pitrey, U. Engelke, M. Barkowsky, R. Pépion, and P. Le Callet, “Subjective quality of SVC-coded videos with different error-patterns concealed using spatial scalability,” in *Proc. 3rd EUVIP Workshop*, Paris, France, 2011, pp. 180–185.
[104] C. Alberti *et al.*, “Automated QoE evaluation of dynamic adaptive streaming over HTTP,” in *Proc. 5th Int. Workshop QoMEX*, Klagenfurt, Austria, 2013, pp. 58–63.
[105] N. Bouten *et al.*, “QoE optimization through in-network quality adaptation for HTTP adaptive streaming,” in *Proc. 8th Int. CNSM*, Las Vegas, NV, USA, 2012, pp. 336–342.

[106] *Sintel, The Durian Open Movie Project*, Blender Foundation, Amsterdam, The Netherlands, 2010.

[107] Y.-F. Ou, Y. Zhou, and Y. Wang, "Perceptual quality of video with frame rate variation: A subjective study," in *Proc. IEEE ICASSP*, Dallas, TX, USA, 2010, pp. 2446–2449.

[108] F. Pereira, "Sensations, perceptions and emotions towards quality of experience evaluation for consumer electronics video adaptations," in *Proc. 1st Int. Workshop VPQM Consum. Electron.*, Scottsdale, AZ, USA, 2005. [Online]. Available: http://enpub.fulton.asu.edu/resp/vpqm/vpqm2005/vpqm05_cover.htm

[109] M. K. Asadi and J.-C. Duford, "Multimedia adaptation by transmoding in MPEG-21," in *Proc. Int. WIAMIS*, Lisbon, Portugal, 2004, p. 83.

[110] M.-N. Garcia and A. Raake, "Parametric packet-layer video quality model for IPTV," in *Proc. 10th Int. Conf. ISSPA*, Kuala Lumpur, Malaysia, 2010, pp. 349–352.

[111] M. Seufert, M. Slanina, S. Egger, and M. Kottkamp, "To pool or not to pool: A comparison of temporal pooling methods for HTTP adaptive video streaming," in *Proc. 5th Int. Workshop QoMEX*, Klagenfurt, Austria, 2013, pp. 52–57.

[112] G. Zhai *et al.*, "Cross-dimensional perceptual quality assessment for low bit-rate videos," *IEEE Trans. Multimedia*, vol. 10, no. 7, pp. 1316–1324, Nov. 2008.

[113] K. Yamagishi and T. Hayashi, "Parametric packet-layer model for monitoring video quality of IPTV services," in *Proc. IEEE ICC*, Beijing, China, 2008, pp. 110–114.

[114] H. Knoche, J. D. Mccarthy, and M. A. Sasse, "How low can you go? The effect of low resolutions on shot types in mobile TV," *Multimedia Tools Appl.*, vol. 36, no. 1/2, pp. 145–166, Jan. 2008.

[115] L. Janowski and P. Romaniak, "QoE as a function of frame rate and resolution changes," in *Proc. 3rd Int. Workshop FMN*, Krakow, Poland, 2010, pp. 34–45.

[116] P. Le Callet, S. Péchard, S. Tourancheau, A. Ninassi, and D. Barba, "Towards the next generation of video and image quality metrics: Impact of display, resolution, contents and visual attention in subjective assessment," in *Proc. 2nd Int. Workshop IMQA*, Chiba, Japan, 2007, pp. 1–9.

[117] R. R. Pastrana-Vidal, J.-C. Gicquel, C. Colomes, and H. Cherifi, "Frame dropping effects on user quality perception," in *Proc. 5th Int. WIAMIS*, Lisbon, Portugal, 2004.

[118] G. Ghinea and J. P. Thomas, "QoS impact on user perception and understanding of multimedia video clips," in *Proc. 6th ACM Int. Conf. Multimedia*, Bristol, U.K., 1998, pp. 49–54.

[119] R. K. Rajendran, M. Van Der Schaar, and S.-F. Chang, "FGS+: Optimizing the joint SNR–temporal video quality in MPEG-4 fine grained scalable coding," in *Proc. IEEE ISCAS*, Scottsdale, AZ, USA, 2002, pp. I-445–I-448.

[120] J.-S. Lee, F. De Simone, and T. Ebrahimi, "Subjective quality evaluation via paired comparison: Application to scalable video coding," *IEEE Trans. Multimedia*, vol. 13, no. 5, pp. 882–893, Oct. 2011.

[121] S. Jumisko-Pyykkö and J. Häkkinen, "Evaluation of subjective video quality of mobile devices," in *Proc. 13th Annu. ACM Int. Conf. Multimedia*, Singapore, 2005, pp. 535–538.

[122] S. Winkler and C. Faller, "Maximizing audiovisual quality at low bitrates," in *Proc. 1st Int. Workshop VPQM Consum. Electron.*, Scottsdale, AZ, USA, 2005.

[123] *Guidelines for Implementation: DASH-HEVC/265 Interoperabolity Points*, DASH Industry Forum, 2013. [Online]. Avaialable: http://dashif.org/w/2013/10/DASH-HEVC-265-base-v0.92.pdf

[124] X. Wang *et al.*, "DASH (HEVC)/LTE: QoE-based dynamic adaptive streaming of HEVC content over wireless networks," in *Proc. IEEE VCIP*, San Diego, CA, USA, 2012. [Online]. Available: http://ieeexplore.ieee.org/xpl/articleDetails.jsp?arnumber=6410858

[125] J. Le Feuvre, J. Thiesse, M. Parmentier, M. Raulet, and C. Daguet, "Ultra high definition HEVC dash data set," in *Proc. 5th Annu. ACM Conf. MMSys*, Singapore, 2014, pp. 7–12.

[126] J. Bankoski *et al.*, "Towards a next generation open-source video codec," in *Proc. IST/SPIE Electron. Imag.—Vis. Inf. Process. Commun. IV*, Burlingame, CA, USA, 2013, pp. 866 606-1–866 606-13.

[127] D. Grois, D. Marpe, A. Mulayoff, B. Itzhaky, and O. Hadar, "Performance comparison of H.265/MPEG-HEVC, VP9H.264/MPEG-AVC encoders," in *Proc. 30th PCS*, San Jose, CA, USA, 2013, pp. 394–397.

[128] O. Verscheure, P. Frossard, and M. Hamdi, "User-oriented QoS analysis in MPEG-2 video delivery," *Real-Time Imag.*, vol. 5, no. 5, pp. 305–314, Oct. 1999.

[129] J. De Vriendt, D. De Vleeschauwer, and D. Robinson, "Model for estimating QoE of video delivered using HTTP adaptive streaming," in *Proc. IFIP/IEEE Int. Symp. IM*, Ghent, Belgium, 2013, pp. 1288–1293.

[130] H. Knoche, J. D. McCarthy, and M. A. Sasse, "Can small be beautiful?: Assessing image resolution requirements for mobile TV," in *Proc. 13th Annu. ACM Int. Conf. Multimedia*, Singapore, 2005, pp. 829–838.

[131] R. T. Apteker, J. A. Fisher, V. S. Kisimov, and H. Neishlos, "Video acceptability and frame rate," *IEEE MultiMedia*, vol. 2, no. 3, pp. 32–40, 1995.

[132] D. Wang, F. Speranza, A. Vincent, T. Martin, and P. Blanchfield, "Toward optimal rate control: A study of the impact of spatial resolution, frame rate, quantization on subjective video quality and bit rate," in *Proc. SPIE VCIP*, Lugano, Switzerland, 2003, pp. 198–209.

[133] T. Hayashi *et al.*, "Effects of IP packet loss and picture frame reduction on MPEG1 subjective quality," in *Proc. Workshop Signal Process.*, Copenhagen, Denmark, 1999, pp. 515–520.

[134] J. Chen and J. Thropp, "Review of low frame rate effects on human performance," *IEEE Trans. Syst., Man, Cybern. A, Syst., Humans*, vol. 37, no. 6, pp. 1063–1076, Nov. 2007.

[135] M. Masry, S. S. Hemami, W. M. Osberger, and A. M. Rohaly, "Subjective quality evaluation of low-bit-rate video," in *Proc. SPIE Photon. West, Electron. Imag.*, San Jose, CA, USA, 2001, pp. 102–113.

[136] J. D. McCarthy, M. A. Sasse, and D. Miras, "Sharp or smooth?: Comparing the effects of quantization vs. frame rate for streamed video," in *Proc. CHI Comput. Syst.*, Vienna, Austria, 2004, pp. 535–542.

[137] N. Van den Ende, H. De Hesselle, and L. Meesters, "Towards content-aware coding: User study," in *Proc. 5th EuroITV Conf.*, Amsterdam, The Netherlands, 2007, pp. 185–194.

[138] Y. Wang, S.-F. Chang, and A. C. Loui, "Subjective preference of spatio-temporal rate in video adaptation using multi-dimensional scalable coding," in *Proc. IEEE ICME*, Taipei, Taiwan, 2004, pp. 1719–1722.

[139] J. Korhonen, U. Reiter, and J. You, "Subjective comparison of temporal and quality scalability," in *Proc. 3rd Int. Workshop QoMEX*, Mechelen, Belgium, 2011, pp. 161–166.

[140] R. Houdaille and S. Gouache, "Shaping HTTP adaptive streams for a better user experience," in *Proc. 3rd Annu. ACM Conf. MMSys*, Chapel Hill, NC, USA, 2012, pp. 1–9.

[141] S. Akhshabi, L. Anantakrishnan, A. C. Begen, and C. Dovrolis, "What happens when HTTP adaptive streaming players compete for bandwidth?" in *Proc. 22nd ACM Int. Workshop NOSSDAV*, Toronto, ON, Canada, 2012, pp. 9–14.

[142] L. De Cicco and S. Mascolo, "An adaptive video streaming control system: Modeling, validation, performance evaluation," *IEEE/ACM Trans. Netw.*, vol. 22, no. 2, pp. 526–539, Apr. 2014.

[143] J. Esteban *et al.*, "Interactions between HTTP adaptive streaming and TCP," in *Proc. 22nd ACM Workshop NOSSDAV*, Toronto, ON, Canada, 2012, pp. 21–26.

[144] T.-Y. Huang, N. Handigol, B. Heller, N. McKeown, and R. Johari, "Confused, timid, unstable: Picking a video streaming rate is hard," in *Proc. ACM Internet Meas. Conf.*, Boston, MA, USA, 2012, pp. 225–238.

[145] A. Mansy, B. Ver Steeg, and M. Ammar, "SABRE: A client based technique for mitigating the buffer bloat effect of adaptive video flows," in *Proc. 4th Annu. ACM Conf. MMSys*, Oslo, Norway, 2013, pp. 214–225.

[146] J. Gettys, "Bufferbloat: Dark buffers in the Internet," *IEEE Internet Comput.*, vol. 15, no. 3, pp. 96–96, May/Jun. 2011.

[147] P. Casas, M. Seufert, S. Egger, and R. Schatz, "Quality of experience in remote virtual desktop services," in *Proc. IFIP/IEEE Int. Symp. IM*, Ghent, Belgium, 2013, pp. 1352–1357.

[148] T. Hoßfeld, R. Schatz, M. Varela, and C. Timmerer, "Challenges of QoE management for cloud applications," *IEEE Commun. Mag.*, vol. 50, no. 4, pp. 28–36, Apr. 2012.

[149] S. Egger, R. Schatz, and S. Scherer, "It takes two to tango—Assessing the impact of delay on conversational interactivity on perceived speech quality," in *Proc. Interspeech*, Makuhari, Japan, 2010, pp. 1321–1324.

[150] A. Raake *et al.*, "IP-based mobile and fixed network audiovisual media services," *IEEE Signal Process. Mag.*, vol. 28, no. 6, pp. 68–79, Nov. 2011.

[151] X. Liu and A. Men, "QoE-aware traffic shaping for HTTP adaptive streaming," *Int. J. Multimedia Ubiquitous Eng.*, vol. 9, no. 2, pp. 33–44, Mar. 2014.

[152] S. Akhshabi, L. Anantakrishnan, C. Dovrolis, and A. C. Begen, "Server-based traffic shaping for stabilizing oscillating adaptive streaming players," in *Proc. 23rd ACM Int. Workshop NOSSDAV*, Oslo, Norway, 2013, pp. 19–24.

[153] M. Ghobadi, Y. Cheng, A. Jain, and M. Mathis, "Trickle: Rate limiting YouTube video streaming," in *Proc. USENIX Conf. ATC*, Boston, MA, USA, 2012, pp. 1–6.

[154] S. Laga *et al.*, "Optimizing scalable video delivery through openflow layer-based routing," in *Proc. IEEE NOMS*, Krakow, Poland, 2014, pp. 1–4.

[155] N. Bouten *et al.*, "Deadline-based approach for improving delivery of SVC-based HTTP adaptive streaming content," in *Proc. IEEE NOMS*, Krakow, Poland, 2014, pp. 1–7.
[156] A. E. Essaili *et al.*, "Quality-of-experience driven adaptive HTTP media delivery," in *Proc. IEEE ICC*, Budapest, Hungary, 2013, pp. 2480–2485.
[157] X. Zhu, Z. Li, R. Pan, J. Gahm, and H. Hu, "Fixing multi-client oscillations in HTTP-based adaptive streaming: A control theoretic approach," in *Proc. IEEE 15th Int. Workshop MMSP*, Pula, Italy, 2013, pp. 230–235.
[158] S. Petrangeli, M. Claeys, S. Latre, J. Famaey, and F. De Turck, "A multi-agent Q-learning-based framework for achieving fairness in HTTP adaptive streaming," in *Proc. IEEE NOMS*, Krakow, Poland, 2014, pp. 1–9.
[159] V. Krishnamoorthi, N. Carlsson, D. Eager, A. Mahanti, and N. Shahmehri, "Helping hand or hidden hurdle: Proxy-assisted HTTP-based adaptive streaming performance," in *Proc. IEEE 21st Int. Symp. MASCOTS*, San Francisco, CA, USA, 2013, pp. 182–191.
[160] P. Georgopoulos, Y. Elkhatib, M. Broadbent, M. Mu, and N. Race, "Towards network-wide QoE fairness using openflow-assisted adaptive video streaming," in *Proc. ACM SIGCOMM Workshop Future Human-Centric Multimedia Netw.*, 2013, pp. 15–20.
[161] R. K. Mok, X. Luo, E. W. Chan, and R. K. Chang, "QDASH: A QoE-aware DASH system," in *Proc. 3rd Annu. ACM Conf. MMSys*, Chapel Hill, NC, USA, 2012, pp. 11–22.
[162] T. Hoßfeld *et al.*, "An economic traffic management approach to enable the triplewin for users, ISPs, overlay providers," in *Towards the Future Internet—A European Research Perspective*, G. Tselentis, J. Domingue, A. Galis, A. Gavras, D. Hausheer, S. Krco, V. Lotz, and T. Zahariadis, Eds. Amsterdam, The Netherlands: IOS Press, 2009, ser. Future Internet Assembly, pp. 24–34.
[163] Internet Engineering Task Force (IETF), ALTO Status Pages. [Online]. Available: http://tools.ietf.org/wg/alto
[164] M. Jarschel, T. Zinner, T. Hoßfeld, P. Tran-Gia, and W. Kellerer, "Interfaces, attributes, use cases: A compass for SDN," *IEEE Commun. Mag.*, vol. 52, no. 6, pp. 210–217, Jun. 2014.
[165] P. Casas, M. Seufert, and R. Schatz, "YOUQMON: A system for on-line monitoring of YouTube QoE in operational 3G networks," *ACM SIGMETRICS Perform. Eval. Rev.*, vol. 41, no. 2, pp. 44–46, Sep. 2013.
[166] S. Traverso *et al.*, "TailGate: Handling long-tail content with a little help from friends," in *Proc. 21st Int. Conf. WWW*, Lyon, France, 2012, pp. 151–160.
[167] M. Seufert *et al.*, "Socially-aware traffic management," in *Social Informatics—The Social Impact of Interactions Between Humans and IT*, K. A. Zweig, W. Neuser, V. Pipek, M. Rohde, and I. Scholtes, Eds. Berlin, Germany: Springer-Verlag, 2014, pp. 25–43.

Michael Seufert received the Diploma degree in computer science in 2011 from the University of Würzburg, Würzburg, Germany, where he is currently working toward the Ph.D. degree. He additionally passed the first state examinations for teaching mathematics and computer science in secondary schools. From 2012–2013, he was with FTW Telecommunication Research Center, Vienna, Austria, working in the area of user-centered interaction and communication economics. He is currently a Researcher at the Chair of Communication Networks, University of Würzburg. His research mainly focuses on QoE of Internet applications, social networks, performance modeling and analysis, and traffic management solutions.

Sebastian Egger received the master's degree in sociology from the University of Graz, Graz, Austria, and the Ph.D. degree in telecommunications from Graz University of Technology, Graz. Since 2010, he has been involved in standardization activities of the ETSI STQ and ITU-T Study Group 12 on Performance, QoS, and QoE. In 2014, he joined the Department of Innovation Systems, AIT Austrian Institute of Technology GmbH, Vienna, Austria, where he is working on technology experience in the domains of human-to-human mediated interaction, interactive services, and HCI. His main research interests are in Quality of Experience for interactive speech, video and web services, as well as HCI support systems for users with special needs.

Martin Slanina received the M.Sc. and Ph.D. degrees in electronics and communication from Brno University of Technology, Brno, Czech Republic, in 2005 and 2009, respectively. He is currently an Assistant Professor at Brno University of Technology. The main focus area of his research is video coding, display, and Quality of Experience in video services, mainly in the context of mobile communication networks.

Thomas Zinner received the Diploma and Ph.D. degrees in computer science from the University of Würzburg, Würzburg, Germany, in 2007 and 2012, respectively. His Ph.D. thesis is on performance modeling of QoE-aware multipath video transmission in the future Internet. He is now the Head of the "Next Generation Networks" Research Group, Chair of Communication Networks, University of Würzburg. His main research interests cover video streaming techniques, implementation of QoE awareness within networks, software-defined networking (SDN) and network virtualization, network function virtualization and the benefits of cloudification, as well as the performance assessment of these technologies and architectures.

Tobias Hoßfeld received the Diploma and Ph.D. degrees in computer science from the University of Würzburg, Würzburg, Germany, in 2003 and 2009, respectively. His professional thesis (habilitation) was on "Modeling and analysis of Internet applications and services." He has been a Professor and Head of the Chair of Modeling of Adaptive Systems, University of Duisburg-Essen, Essen, Germany, since 2014. During the time of this work, he was heading the "Future Internet Applications and Overlays" Research Group, Chair of Communication Networks, University of Würzburg. He has published more than 100 research papers in major conferences and journals, receiving four best conference paper awards, three awards for his Ph.D. thesis, and the Fred W. Ellersick Prize 2013 from the IEEE Communications Society for one of his articles on QoE.

Phuoc Tran-Gia is a Professor and Director of the Chair of Communication Networks, University of Würzburg, Würzburg, Germany. He is also a member of the Advisory Board of Infosim (Germany) specialized in IP network management products and services. He is also a Cofounder and Board Member of Weblabcenter, Inc. (Dallas, TX), specialized in crowdsourcing technologies. He received the Diploma and Ph.D. degrees in electrical engineering from the University of Stuttgart, Germany, in 1977, and from the University of Siegen, Germany, in 1982, respectively, and was at industries at Alcatel (SEL) and IBM Zurich Research Laboratory. He is active in several EU framework projects and COST actions. He was the Coordinator of the German-wide G-Lab Project "National Platform for Future Internet Studies" aiming to foster experimentally driven research to exploit future Internet technologies. He has published more than 100 research papers in major conferences and journals. His research activities focus on performance analysis of the following major topics: Future Internet and Smartphone Applications; QoE Modeling and Resource Management; Software Defined Networking and Cloud Networks; Network Dynamics and Control; and Crowdsourcing. Prof. Tran-Gia was a recipient of the Fred W. Ellersick Prize 2013 from the IEEE Communications Society.