A Generic Approach to Video Buffer Modeling Using Discrete-Time Analysis

VALENTIN BURGER, THOMAS ZINNER, LAM DINH-XUAN, FLORIAN WAMSER, and PHUOC TRAN-GIA, University of Würzburg

The large share of traffic in the Internet generated by video streaming services puts high loads on access and aggregation networks, resulting in high costs for the content delivery infrastructure. To reduce the bandwidth consumed while maintaining a high playback quality, video players use policies that control and limit the buffer level by using thresholds for pausing and continuing the video download. This allows shaping the bandwidth consumed by video streams and limiting the traffic wasted in case of playback abortion. Especially in mobile scenarios, where the throughput can be highly variant, the buffer policy can have a high impact on the probability of interruptions during video playback. To find the optimal setting for the buffer policy in each network condition, the relationship between the parameters of the buffer policy, the network throughput dynamics, and the corresponding video playback behavior needs to be understood. To this end, we model the video buffer as GI/GI/1 queue with *pq*-policy using discrete-time analysis. By studying the stochastic properties of the buffer-level distribution, we are able to accurately evaluate the impact of network and video bitrate dynamics on the video playback quality based on the buffer policy. We find a fundamental relationship between the bandwidth variation and the expected interarrival time of segments, meaning that overproportionately more bandwidth is necessary to prevent stalling events for high bandwidth variation. The proposed model further allows to optimize the trade-off between the traffic wasted in case of video abortion and video streaming quality experienced by the user.

CCS Concepts: • **Information systems → Multimedia streaming**; • **Networks → Network performance analysis**; *Network performance modeling*; *Network measurement*;

Additional Key Words and Phrases: Adaptive video streaming, quality of experience, buffering policy, bandwidth variation, performance analysis

ACM Reference format:
Valentin Burger, Thomas Zinner, Lam Dinh-Xuan, Florian Wamser, and Phuoc Tran-Gia. 2018. A Generic Approach to Video Buffer Modeling Using Discrete-Time Analysis. *ACM Trans. Multimedia Comput. Commun. Appl.* 14, 2s, Article 33 (April 2018), 23 pages.
https://doi.org/10.1145/3183511

1 INTRODUCTION

With high-resolution mobile devices and live streaming services, video streaming continues to be the service generating the largest share of traffic in the Internet (Cisco 2017). Since small

This work was partly funded by Deutsche Forschungsgemeinschaft (DFG) under grants ZI 1334/2-1 and TR 257/43-1. The authors alone are responsible for the content of the article.

Authors' addresses: V. Burger (corresponding author), T. Zinner, L. Dinh-Xuan, F. Wamser, and P. Tran-Gia, University of Würzburg, Institute of Computer Science, Würzburg, Germany; emails: {valentin.burger, zinner, lam.dinh-xuan, florian.wamser, trangia}@informatik.uni-wuerzburg.de.

https://doi.org/10.1145/3183511

degradation in the perceived video streaming quality can highly impact the acceptance of the service (Fiedler et al. 2010), video streaming services have high requirements on the network transporting the video data and on the quality of experience (QoE), which corresponds to the quality of the service subjectively perceived by the end user while watching.

Main influence factors on the video streaming QoE are video interruptions that occur if the video buffer runs empty, which we refer to as stalling events. A high video playback quality without playback interruptions can be ensured if enough bandwidth is available to download each video segment in time. HTTP adaptive streaming (HAS) accounts for bandwidth variations by encoding the video in different quality levels. This allows adapting the quality level and thus the video bitrate to the available bandwidth to prevent the video from stalling. Especially in mobile scenarios, where the bandwidth is highly variable, low buffer levels can have a high impact on the playback quality, as enough playtime has to be buffered to avoid an empty buffer in low bandwidth periods. James et al. (2016) find that by using multipath TCP, the bandwidth variation has a higher impact on video streaming QoE than the average bandwidth, showing that an unstable secondary path can harm the user experience when added to a stable primary path.

In addition to adapting the quality level, a high buffer level allows compensating variations of the available network capacities. However, if the viewer aborts the playback or seeks to a later position in the video, the bandwidth consumed to download the video material that is not played out is wasted. Hence, there is a trade-off between keeping the buffer level high to avoid playback interruptions and keeping the buffer level low to reduce the traffic wasted in case of seeking or playback abortion. To control the buffer level, current video players implement buffer policies that try to keep the buffer level in a range between two thresholds p and q (Ramos-Muñoz et al. 2014; Wamser et al. 2016), which pause and continue the buffering process.

To find the optimal setting of the buffer policy that achieves the best trade-off in each network condition, the relationship between the parameters of the buffer policy, the network dynamics, and the video playback needs to be understood. Existing work (Hoßfeld et al. 2015) does not allow analyzing the performance of video buffers for generally distributed bandwidth and bitrate distributions to consider varying network and video dynamics. Furthermore, existing work (Luan et al. 2010; Moldovan and Hoßfeld 2016) only considers the server policy, which controls when the video player starts and stops the playback. The buffer policy, which controls the buffer level by determining when to request video segments, is not considered.

To close this gap, we develop a generic approach to model a video buffer as a GI/GI/1 queue with pq-policy. Using discrete-time analysis, we calculate the probability distribution of unfinished work in the system, which corresponds to the buffer-level distribution of the video buffer. This allows studying stochastic properties of the buffer-level distribution and to evaluate the impact of bandwidth and bitrate variation on the video playback quality based on the buffer policy.

The contribution of this article is fourfold. First, we provide a general model for video buffer analysis with pq-policy. Second, we show the applicability of our model to HAS systems by conducting measurements for all relevant influence factors. Third, we find a fundamental relationship between the bandwidth variation and the expected interarrival time of segments, meaning that for increasing bandwidth variation, overproportionately more bandwidth is necessary to prevent stalling events. Finally, we show how the policy thresholds p and q can be used to adjust the trade-off between smooth video playback and bandwidth consumption based on the network conditions.

The rest of the work is structured as follows. Background and related work referring to video streaming quality and analysis of video buffer policies is described in Section 2. The discrete-time analysis method to model a GI/GI/1 buffer with pq-policy is presented in Section 3. The testbed setup and the measurements are described in Section 4. Numerical examples derived by analysis are given in Section 5, and we present our conclusion in Section 6.

2 BACKGROUND AND RELATED WORK

The importance of HTTP video streaming services has driven intensive research activities on how to optimize video delivery while also maintaining a high QoE. We first show methods to estimate the quality perceived by users watching HTTP video streams based on network parameters and then present work related to the analysis of queuing systems with one server, which can be used for video buffer analysis.

2.1 User-Perceived Quality of HTTP Adaptive Video Streams

In telecommunication networks, the quality of service (QoS) is described objectively by network parameters such as packet loss, delay, or jitter. A good QoS does not necessarily mean that all customers experience a good service quality. The concept of QoE (Le Callet et al. 2012) explicitly refers to the user-perceived quality by relying on subjective criteria.

Since TCP provides reliable transmission of the video data, typically no video distortions or artifacts can be observed, but resource problems in the network or at the video server result in waiting times of the video during playout. To monitor and estimate the QoE, the QoS network parameters and other measurable metrics, such as waiting times or quality adaptations, are mapped by QoE models.

According to Hoßfeld et al. (2012), waiting times can be classified into initial delay, i.e., waiting times before video playback, and stalling events, i.e., interruptions during video playback. For HTTP video streaming, the key influence factors on QoE are initial delay and stalling events (Hoßfeld et al. 2011, 2013a). The subjective studies conducted in Hoßfeld et al. (2013b) show a high negative impact of more stalling events for the same overall stalling time on the QoE. The IQX hypothesis (Fiedler et al. 2010) states that the change of QoE depends on the current level of QoE, given the same amount of change of the QoS value. This results in an exponential decay of the QoE. An exponential relation between the QoE of video streams and the duration and number of stalling events is derived in Hoßfeld et al. (2013a) for 30s videos. Hence, the IQX hypothesis provides an accurate model for QoE estimation, which is discussed in detail in Section 3.2.

HAS introduces quality adaptations during video playback as an additional influence factor and allows to trade-off interruptions and adaptations. Conventional rate adaptation algorithms (Li et al. 2014) are buffer aware and use reactive bandwidth estimation to select the quality level for the next segment request. In Yang et al. (2011), an underflow probability estimation model based on large deviation theory relying not only on the buffer level but also on its variation is developed for rate adaption. Bao and Valentin (2015) find that rate adaptation algorithms cannot effectively react to channel variation by adapting the video bitrate only to the state of the playback buffer and develop an optimal solution that adapts to the current and predicted channel state. In Yang et al. (2016), a rate adaption algorithm is proposed that adapts the bitrate to the characteristics of variable bitrate downlink channels of LTE systems to enable smooth adaptation in dynamic network conditions.

In Hoßfeld et al. (2014), the influence factors of the adaptation parameters for HAS with two quality layers are investigated, showing that the time on the highest video quality layer has a significant impact on the QoE, and the number of quality switches can be neglected. Liu et al. (2015) consider how the stalls are distributed together with the amount of motion in the video and derive impairment functions for initial delay, stalling events, and quality layer fluctuations. However, according to Seufert et al. (2015), reducing the duration and number of stalling events has higher priority than reducing other disturbances such as longer initial delays or video quality adaptation. Therefore, we focus on the duration and number of stalling events as main influence factors of video streaming QoE in this work. We apply the generalized QoE model (Hoßfeld et al. 2015) to assess the QoE based on the duration and number of stalling events.

2.2 Analysis of Video Buffer Policies

To classify work related to the analysis of video buffer policies, we need to distinguish buffer policies from server policies. We first explain the difference between server and buffer policies and then provide related work for each of these policies with respect to video buffering.

To save resources in production systems, the server is shut down if only few or no jobs are in the system. The server policy determines if and for how long a server is shut down. Considering a video service, the video player corresponds to the server and the video buffer corresponds to the queue of the queuing system (Figure 1). Hence, in case of a video buffer, shutting down the server means stopping the playback, which corresponds to a stalling event. To avoid an empty buffer, video players use buffer policies to control the buffer level. The buffer policy determines when new jobs, which correspond to segments in case of a video buffer, are requested.

According to Zhang et al. (2005), three types of server policies have been discussed in the literature. The (ν,N)-policy determines the state of the server dependent on the number of jobs in the queue. In the (ν,N)-policy (Yadin and Naor 1967), the server is turned on if N jobs are in the queue and shut down if ν jobs are in the queue. Similarly, the D-policy (Balachandran and Tijms 1975; Boxma 1976) determines the state of the server dependent on the unfinished work in the system. In the T-policy, studied for a M/G/1-T queue in Heyman (1977), the server is activated T time units after the end of the last busy period. The (ν,N)-policy and the D-policy are both suitable for video playback. If the (ν,N)-policy is used with $\nu = 0$, the player stops playback if the buffer runs empty and continues the playback as soon as N segments are in the buffer. The D-policy works similarly, with the difference that the playback continues as soon as the number of buffered segments is equivalent to the playback time defined by D.

Hoßfeld et al. (2015) investigate the trade-off between initial delay and stalling of video streams and the impact on QoE for a M/M/1 queue with the D-policy based on a mean value analysis. In Böhm and Mohanty (1993), the transient solution of M/M/1 queues under the D-policy is derived by a combinatorial method. The applicability of these models are limited for video buffers, as they assume Poisson arrivals of the segments and exponentially distributed playback durations of the video segments. The assumptions are very strong, as the playback duration of segments is fixed on most platforms and Poisson arrivals do not allow modeling different bandwidth conditions.

To increase the parameter space to also consider the variation of network bandwidth and video bitrates, Moldovan and Hoßfeld (2016) investigate a GI/GI/1-D system by simulation. This allows investigating the impact of variation of the network bandwidth on the QoE in video streaming. The analysis is limited to simulation results and only covers the D-policy of the player. The buffer policy is not considered, which is prevalent in video players.

In Luan et al. (2010), the impact of network dynamics on the user-perceived video quality is investigated by modeling the playback buffer at the receiver as a GI/GI/1/N queue with limited buffer size N using diffusion approximation. The video quality is determined based on the initial delay and the fluency of video playback. In this case, blocking jobs because of the limited buffer size corresponds to pausing the buffering process. The buffer policy of current video players does not just limit the buffer size but uses two thresholds to pause and continue the buffering process.

To consider the buffer policy with continue and pause buffering threshold p and q, we develop a generic approach to model a GI/GI/1 queue pq-buffer policy using discrete-time analysis. The discrete-time analysis technique for a GI/GI/1 queue with bounded delay has been developed in

Tran-Gia (1993), which serves as basis for the analysis in this work. Note that setting $p = q = \mathrm{N}$ corresponds to the GI/GI/1/N queue with limited buffer size N.

3 MODEL AND ANALYSIS

Modern video streaming systems are based on a segmentation of the video data to enable different quality levels and adaptive streaming. When streaming a video, the video segment arrivals at the player generate a discrete sequence of events. This property allows using discrete-time modeling techniques for performance evaluation of the video streaming systems.

For simplicity, we make four core assumptions on the arrival and service rates of the video segments. These four assumptions are stated and discussed in the following.

Assumption 1. *The interarrival times of segments are independent.*

We assume that the interarrival times of segments, which depend on the segment duration, the video bitrate of the segment, and the current network conditions, are independently distributed. In practice, the interarrival times of segments may mutually depend on each other. Lengthy scenes with slowly moving images require less information to be encoded, as codecs can only transmit the residuals and thus only require a low bitrate. Frequently changing scenes require a high bitrate to encode the information of the different images. Therefore, the bitrate of a sequence of video segments is correlated, dependent on the dynamics of the video content. The bitrate of a segment determines its data volume and thus its download time, which corresponds to its interarrival time.

In case of HAS, the different quality layers have a high impact on the bitrate of the segments. A switch to a higher-quality layer having a significantly higher bitrate has a strong impact on the download time and consequently on the arrival time of the next segment.

It is assumed that the selection of quality levels are independent for each segment. Different quality layers and their selection by HAS logics are reflected, as the model allows considering general bitrate distributions that allow specifying the mean and variance of the bitrate.

A high number of adaptation logics are used in practice, which may, e.g., depend on the buffer level, the available bandwidth (Adzic et al. 2011; Stockhammer 2011), or even the signal strength and the channel state of the mobile link (Bao and Valentin 2015; Mekki et al. 2017; Xie et al. 2015). By conditioning the bitrate of a segment on the available bandwidth or the buffer level, our model can be extended to consider a specific adaptation logic. To provide a generic model, the dependence of the bitrate of subsequent segments due to a quality switch triggered by the adaptation logic is not considered.

We show the applicability of the proposed model to compute influence factors such as stalling frequency and duration by comparing the model outcome with measurements.

Assumption 2. *The service times of segments are independent.*

We assume that the service time of a segment, corresponding to its playtime, is independent of the playtime of the previous segment. Typical HAS systems employ a fixed segment playtime, i.e., with exception of the last segment, each segment provides the same amount of video playback time. The model is general, allowing to consider varying segment playtimes, which covers fixed segment playtimes. In case of fixed segment playtimes, the service times are constant and thus independent from each other.

Assumption 3. *Only complete segments are available for playback.*

We assume that only segments that are downloaded completely are available for playback. If the segment encoding allows partial decoding of video frames, players may be able to play partial

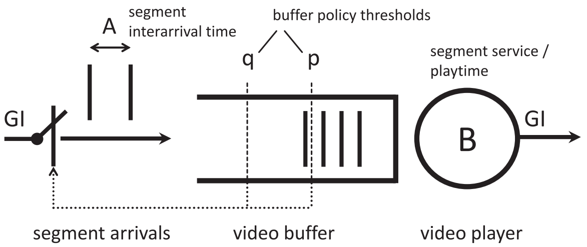


Fig. 1. Queuing model of a GI/GI/1 video buffer with pq-policy.

segments as they arrive from the socket buffer. The corresponding effects, namely that stalling durations and frequencies might be reduced to a certain extent, are not part of the analysis conducted in this work.

Assumption 4. *The protocol overhead is zero.*

We assume that the network overhead of transport and application layer protocols caused by headers and congestion control is negligible. The effective bandwidth available for video transmission is thus lower than the available bandwidth. In practice, this can lead to inaccurate approximations, especially if the segment size is small, which increases the weight of the overhead. To derive accurate approximations by the model in cases where the protocol overhead is not negligible, the effective bandwidth can be used instead of the configured bandwidth to calculate the segment interarrival time distribution. The effective bandwidth is derived by the download time of a segment divided by its file size.

Based on these assumptions, we model the video buffer as a GI/GI/1 queuing system. To consider the buffer policy implemented by current video players, we consider two thresholds p and q that control the buffer level. The thresholds pq define a hysteresis that modulate the buffering by an on-off process, where q is the maximum buffer level in seconds. If q is exceeded, the buffering is paused, inducing an off-period that lasts until the buffer drops below p. The queuing model of the GI/GI/1 video buffer with pq-policy is depicted in Figure 1.

3.1 The GI/GI/1 Queue With pq-policy

In this section, we introduce a GI/GI/1 queue with pq-policy. If the buffer level in seconds playtime exceeds q upon a segment arrival, the segment requests are paused by the video player until the buffer drops below a lower threshold p.

The modification of the basic GI/GI/1 system follows (Tran-Gia 1993), where the discrete-time analysis method is described in detail and used to model a GI/GI/1 queue with bounded delay.

We use the following random variables (RVs) for modeling the video buffer:

- U_n: RV for the buffer level immediately upon arrival of segment $(n-1)$
- A_n: RV for the interarrival time of segment n
- B_n: RV for the service/playtime of segment n
- C_n: RV for the average bitrate of segment n
- D_n: RV for the average throughput received for downloading segment n
- RTT_n: RV for the round trip of the request for segment n.

The respective probability mass functions are denoted by $u_n(k)$, $a_n(k)$, and $b_n(k)$. A snapshot of the state process development in the system is shown in Figure 2. The current video playback buffer level, which corresponds to the unfinished work in case of a GI/GI/1 queue, immediately upon the $(n-1)$-th segment arrival, is denoted by U_n. Assuming that the video player requests

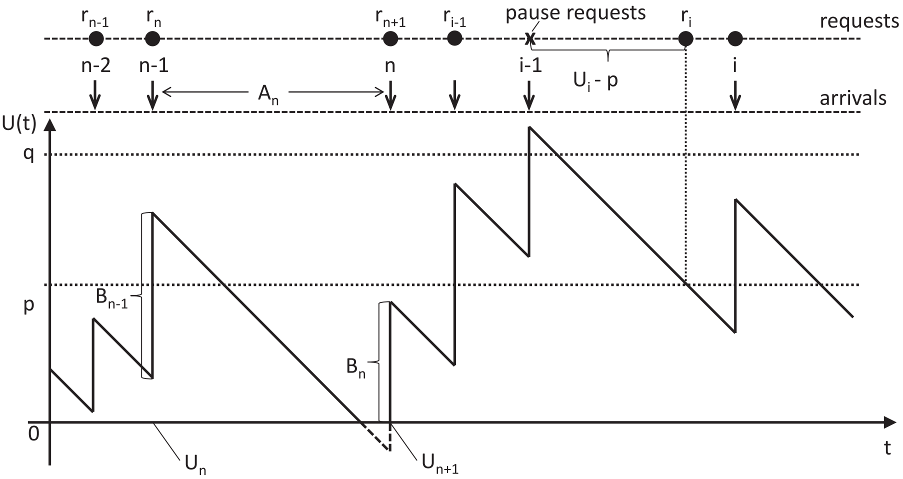


Fig. 2. Sample state process of a GI/GI/1 buffer with pq-policy.

the next segment n immediately upon arrival of the $(n-1)$-th segment, the interarrival time of the segment A_n equals the sum of the time to request the segment and the time to download the segment. Upon arrival of segment n, its playtime B_n is added to the buffer level.

Given the playtime of the segment B_n, the download time of the segment depends on its bitrate C_n and the download throughput D_n. Considering the round-trip time of the request RTT_n, the interarrival time of segment n is

$$A_n = RTT_n + \frac{C_n \cdot B_n}{D_n}. \tag{1}$$

Note that the distribution of A_n has to be calculated by a ratio distribution to consider the bitrate of the segment and the download bandwidth. The mean of the ratio distribution of two generally distributed RVs can be approximated by a Taylor expansion (Benaroya et al. 2005). If RTT_n is negligible, $E[A_n]$ is approximated by

$$E[A_n] = E\left[\frac{C_n \cdot B_n}{D_n}\right] \approx \frac{E[C_n \cdot B_n]}{E[D_n]} - \frac{COV[C_n \cdot B_n, D_n]}{E[D_n]^2} + \frac{E[C_n \cdot B_n]}{E[D_n]^3} VAR[D_n]. \tag{2}$$

Details on the calculation of the ratio distribution of two RVs are shown in Section A of the Appendix.

The service rate of the video player is determined by the video bitrate that corresponds to the amount of video data played out per time unit, e.g., bits per second. Hence, the average bitrate corresponds to the service rate of video playtime by the video player, and thus we denote $E[C_n] = \mu$. The average throughput corresponds to the arrival rate of video playtime in the queue, and thus we denote $E[D_n] = \lambda$. In the parameter studies, we consider the bandwidth provisioning factor a, i.e., the offered load on the video player, which corresponds to the bandwidth to bitrate ratio:

$$a = \frac{E[D_n]}{E[C_n]} = \frac{\lambda}{\mu}. \tag{3}$$

A fluent video playback is generally only possible if $a > 1$. If more video data is delivered than is played out, the video buffer will increase on average until the pause buffering threshold q is reached. In this case, the bandwidth available for delivering the video data is higher than the average video bitrate, which means that the bandwidth provisioning factor is $a > 1$. If the bandwidth variation is zero, this case is trivial, as the buffer level will always increase and no stalling events will occur. Since our model allows studying the bandwidth variation as a parameter, the case $a > 1$ can be investigated. Thus, it is important to understand the implications of different degrees of bandwidth variation on the video streaming behavior.

If the average bandwidth is lower than the average video bitrate ($a < 1$), it means that the buffer level will decrease in the long term. Eventually the buffer will drop to zero. Considering this case is also relevant for practice, as it corresponds to situations where the bandwidth is low. This is, e.g., an issue in mobile environments due to signal strengths. Here, studying the impact of bandwidth variations is also important, as it has a high impact on the duration and frequency of stalling events, and allows to accurately estimate the severity of the quality degradation.

To derive the discrete time model of the buffer level in the GI/GI/1 queue with pq-policy, we introduce the following notations. Observing the $(n-1)$-th segment in the system and the condition for a buffering pause upon arrival instant, the following conditional RVs for the workload seen by an arriving segment immediately upon its arrival are introduced:

$$U_{n,0} = U_n | U_n < q, \tag{4}$$

$$U_{n,1} = U_n | U_n \geq q, \tag{5}$$

$$U_{n+1,0} = U_{n+1} | U_n < q, \tag{6}$$

$$U_{n+1,1} = U_{n+1} | U_n \geq q. \tag{7}$$

The normalized distributions of the RVs are

$$u_{n,0}(k) = \frac{\sigma^q[u_n(k)]}{P(U_n < q)} = \frac{\sigma^q[u_n(k)]}{\sum_{i<q} u_n(i)}, \tag{8}$$

$$u_{n,1}(k) = \frac{\sigma_q[u_n(k)]}{P(U_n \geq q)} = \frac{\sigma_q[u_n(k)]}{\sum_{i\geq q} u_n(i)}, \tag{9}$$

where $\sigma^m(\cdot)$ and $\sigma_m(\cdot)$ are operators that truncate parts of a probability distribution above or below a threshold m. The resulting unnormalized pseudodistributions are defined by

$$\sigma^m[x(k)] = \begin{cases} x(k), & k < m \\ 0, & k \geq m, \end{cases} \tag{10}$$

$$\sigma_m[x(k)] = \begin{cases} 0, & k < m \\ x(k), & k \geq m. \end{cases} \tag{11}$$

To describe the processes where the video player stops playback to rebuffer and when the player pauses segment requests, the sweep operators $\pi_{m,\mathrm{D}}(\cdot)$ and $\pi^{m,\mathrm{D}}(\cdot)$ are defined as follows (if parameter D is omitted, $\mathrm{D} = m$):

$$\pi_{m,\mathrm{D}}[x(k)] = \begin{cases} x(k), & k \geq m, k \neq \mathrm{D} \\ x(\mathrm{D}) + \sum_{i<m} x(i), & k = \mathrm{D} \\ 0, & k < m, \end{cases} \tag{12}$$

$$\pi^{m,\mathrm{D}}[x(k)] = \begin{cases} 0, & k > m \\ x(D) + \sum_{i\geq m} x(i), & k = D \\ x(k), & k \leq m, k \neq \mathrm{D}. \end{cases} \tag{13}$$

The following equations describe the dynamics of the buffer level based on the unfinished work immediately upon segment arrival U_n. The $\Pi_{m,\mathrm{D}}(X)$ operator manipulates the RV X according to the $\pi_{m,\mathrm{D}}(\cdot)$ sweep operator:

$$\Pi_{m,\mathrm{D}}(X) = \begin{cases} X, & X \geq m \\ D, & X < m. \end{cases} \tag{14}$$

(1) $U_n < q$: request segment

$$U_{n+1,0} = \Pi_{0,\mathrm{D}}(U_{n,0} - A_n) + B_n \tag{15}$$

$$u_{n+1,0}(k) = \pi_{0,\mathrm{D}}[u_{n,0}(k) * a_n(-k)] * b_n(k) \tag{16}$$

(2) $U_n \geq q$: pause segment requests for $(U_n - p)$

$$\begin{aligned} U_{n+1,1} &= \Pi_{0,\mathrm{D}}(U_{n,1} - (U_{n,1} - p) - A_n) + B_n \\ &= \Pi_{0,\mathrm{D}}(p - A_n) + B_n \end{aligned} \tag{17}$$

$$u_{n+1,1}(k) = \pi_{0,\mathrm{D}}[\pi^p(u_{n,1}(k)) * a_n(-k)] * b_n(k) \tag{18}$$

The operator $*$ points to a convolution of the respective probability distributions. From the preceding equations, we obtain a recursive relation to calculate the buffer level at segment arrivals.

$$\begin{aligned} u_{n+1}(k) &= \pi_{0,\mathrm{D}}[\sigma^q(u_n(k)) * a_n(-k)] * b_n(k) + \pi_{0,\mathrm{D}}[\pi^p(\sigma_q(u_n(k))) * a_n(-k)] * b_n(k) \\ &= \pi_{0,\mathrm{D}}\Big[\Big(\sigma^q(u_n(k)) + \pi^p(\sigma_q(u_n(k)))\Big) * a_n(-k)\Big] * b_n(k) \end{aligned} \tag{19}$$

An iterative procedure is implemented according to this equation to calculate the buffer level upon segment arrivals. The iterative procedure can be used for stationary and nonstationary conditions, where the indices n and $(n+1)$ can be suppressed in the equation under stationary conditions. The stationary conditions correspond to a video with infinite duration. Figure 3 depicts the computational diagram of the iterative procedure. The buffer-level distribution in step n $u_n(k)$ is conditioned on the buffering threshold q by the σ operators. In case of a buffer level higher than q, buffering is paused until threshold p is reached corresponding to the π^p operator. During the download time for the next segment, the buffer is reduced by $a_n(k)$, which can result in a buffer deficit considered in the buffer-level distribution with negative buffer levels $\hat{u}_n(k)$. In case of negative buffer levels, the playback is paused for the buffer deficit and playback is resumed until the first segment arrival that exceeds the initial buffer threshold D, realized by the $\pi_{0,\mathrm{D}}$ operator. On arrival, the segment playtime $b_n(k)$ is added to the buffer, resulting in the buffer level immediately upon segment arrival of segment n, $\hat{u}_{n+1}(k)$. Since the buffering process can be composed by modular operators, this approach is generic. The computational diagram could be easily adapted or extended to model a modified buffering process.

We calculate the buffer-level distribution with negative buffer levels $\hat{u}_n(k)$, i.e., before the $\pi_{0,\mathrm{D}}$ operator, to obtain performance measures:

$$\hat{u}_n(k) = \Big(\sigma^q(u_n(k)) + \pi^p(\sigma_q(u_n(k)))\Big) * a_n(-k). \tag{20}$$

The negative buffer level corresponds to the time between the buffer running empty and playback stops until the next segment arrives and playback continues. Thus, it gives us the duration of a stalling event. The actual buffer is never negative, as the probability mass of the negative buffer level is added to the empty buffer level by the π_0 operator.

Figure 4 shows an example of the probability mass of the buffer-level distribution with negative buffer levels $\hat{u}_n(k)$. The buffer level k is depicted on the x-axis and the corresponding probability mass on the y-axis. The dashed lines indicate the empty buffer and the buffer levels of the continue buffering threshold p and the pause buffering threshold q. The probability mass of negative buffer

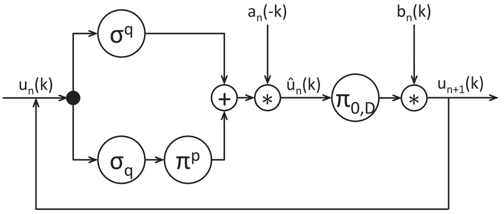


Fig. 3. Computational diagram for a GI/GI/1 buffer with pq-policy.

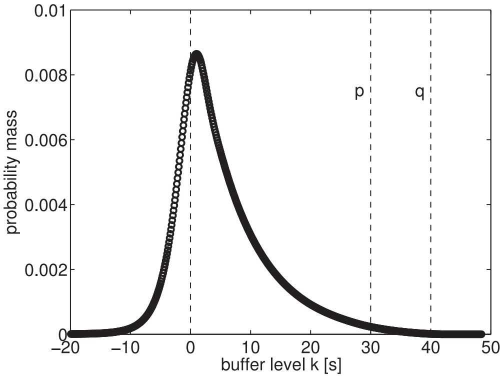


Fig. 4. Example probability mass of the buffer-level distribution with negative buffer levels $\hat{u}_n(k)$.

levels down to −20s is nonzero, which means that stalling events with a duration of up to 20s occur with the corresponding probability. The sum of the probability mass of negative buffer levels gives the stalling probability. The buffer policy thresholds limit the positive buffer level resulting in a probability mass close to zero for buffer levels greater than q.

Negative buffer levels are in fact not possible, but they give us the time until the next segment arrival. Thus, we can calculate different performance measures from the buffer-level distribution with negative buffer levels $\hat{u}_n(k)$. Under stationary conditions, the probability that the buffer runs empty and the video stalls or freezes, i.e., the stalling probability, is

$$p_{st} = \sum_{i<0} \hat{u}(i). \tag{21}$$

In case of a stalling event, the stalling duration in stationary conditions is the time the playback is paused to rebuffer the next segment to resume playback:

$$L = -\sum_{i<0} i \cdot \hat{u}(i). \tag{22}$$

The average buffer level $\bar{u}$, which corresponds to the traffic wasted in case of abortion of playback, is derived by calculating the average of the buffer level upon arrival $u(i)$ and prior to the next arrival or stall $\pi_{0,\mathrm{D}}[\hat{u}(i)]$. To account for the empty buffer during stalling events, the sum is weighted by $\frac{B}{B+L}$, where B is the average playtime of a segment and L the average stalling duration:

$$\bar{u} = \frac{1}{2}\frac{B}{B+L}\sum_i i \cdot (u(i) + \pi_0[\hat{u}(i)]). \tag{23}$$

For nonstationary cases, a total number of N segments is downloaded. If the playback starts after the first segment has arrived, the subsequent $N-1$ segments can arrive late and cause a stall. This leads to the following formula for the overall stalling probability in case of N segments:

$$p_{st,N} = \frac{1}{N-1}\sum_{n=2}^{N}\sum_{i<0} \hat{u}_n(i). \tag{24}$$

The rate of stalling events in a video with N segments is derived by

$$\phi_N = \frac{p_{st,N}}{E[B_n]} = \frac{p_{st,N}N}{T}. \tag{25}$$

The average stalling duration in nonstationary conditions is accordingly calculated by

$$L_N = -\frac{1}{N-1}\sum_{n=2}^{N}\sum_{i<0} i \cdot \hat{u}_n(i). \tag{26}$$

The average buffer level in the transient phase, which corresponds to the average traffic wasted in case of abortion of playback, is calculated by considering the buffer level upon arrival and prior to the next arrival or stall. To account for the empty buffer during stalling events, the sum is weighted by $\frac{T}{T+(N-1)L_N}$:

$$\bar{u}_N = \frac{1}{2}\frac{T}{T+(N-1)L_N}\frac{1}{N-1}\sum_{n=2}^{N}\sum_{i} i \cdot (u_n(i) + \pi_0[\hat{u}_n(i)]). \tag{27}$$

Based on the duration and number of stalling events, a QoE value can be calculated that assesses the quality perceived by a user watching the video stream.

3.2 QoE Model

The IQX hypothesis (Fiedler et al. 2010) states that the change of QoE depends on the current level of QoE, given the same amount of change of the QoS value. This results in an exponential decay of the QoE. An exponential relation between the QoE of video streams and the number and duration of stalling events is derived in Hoßfeld et al. (2013a) for 30s videos. The relation is generalized for different user profiles with the duration parameter α and the interruption parameter β in Hoßfeld et al. (2015). We use the generalized form of the equation to calculate the QoE value Q_1 based on the average duration L and average number of stalling events $p_{st}N = \phi T$.

$$Q_1 = e^{-(\alpha L+\beta)p_{st}N} = e^{-(\alpha L+\beta)\phi T} \tag{28}$$

Despite video interruptions caused by stalling events, the initial delay is another main influence factor on the video QoE. We use the relation derived in Hoßfeld et al. (2012), which is generalized in Hoßfeld et al. (2015) to describe the impact of the initial delay T_0 on the QoE. The sensitivity to initial delay is described by the initial delay parameter γ. The shape of the function is set by the parameter c, which is set to $c = 5.381$ according to the study in Hoßfeld et al. (2012).

$$Q_2 = -\gamma \log_{10}(T_0 + c) + \gamma \log_{10}(c) + 1 \tag{29}$$

In the combined model $Q_1 \cdot Q_2$, the QoE is reduced according to the initial delay by Q_2 and then further reduced by the exponential decay in Q_1 according to the duration and number of stalling events. It is shown in Hoßfeld et al. (2016) that the multiplicative model preserves the IQX hypothesis and properly captures results from the literature. According to Hoßfeld et al. (2016), the values derived from Q_1, Q_2, and $Q_1 \cdot Q_2$ can be scaled from 1 to 5 to allow reading in terms of mean opinion scores (MOS) on a 5-point ACR scale (ITU-T RECOMMENDATION 1999), which correspond to (1) bad, (2) poor, (3) fair, (4) good, and (5) excellent quality.

Table 1 gives the notation of the article, describing the parameters and providing default values. Default values were taken from streaming systems like metacafe.com, which is investigated in more detail in Section 4. An initial buffering threshold of $D = 0s$ implies that the playback starts initially and after a stall as soon as the next segment is downloaded. If not stated differently in the parameter studies, the parameters are set to the default values. The bandwidth and bitrate variability is defined per segment in accordance to the discrete-time events. The parameters α, β, and γ specify the QoE model.

Table 1. Notation of the Article

Parameter	Description	Default
T	Video playtime	240s
N	Number of segments	24
a	Bandwidth provisioning factor	1.2
λ	Average bandwidth	600kbps
c_λ	Coefficient of variation (CoV) of bandwidth	0.2
μ	Average bitrate	500kbps
c_μ	CoV of bitrate	0.1
p	Continue buffering threshold	30s
q	Pause buffering threshold	40s
D	Initial buffering threshold	0s
α	Duration parameter	0.15
β	Interruption parameter	0.2
γ	Initial delay parameter	0.3

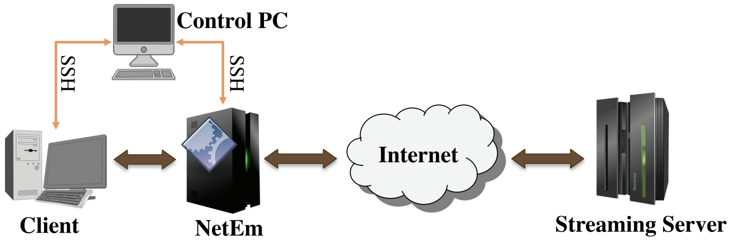


Fig. 5. Overview of measurement setup.

4 MEASUREMENT DESCRIPTION AND APPLICABILITY OF THE MODEL

To investigate the impact of the buffering threshold in real environments and to validate the analytic model and show its applicability, we set up a testbed for measurements with an Internet service and a controlled artificial environment.

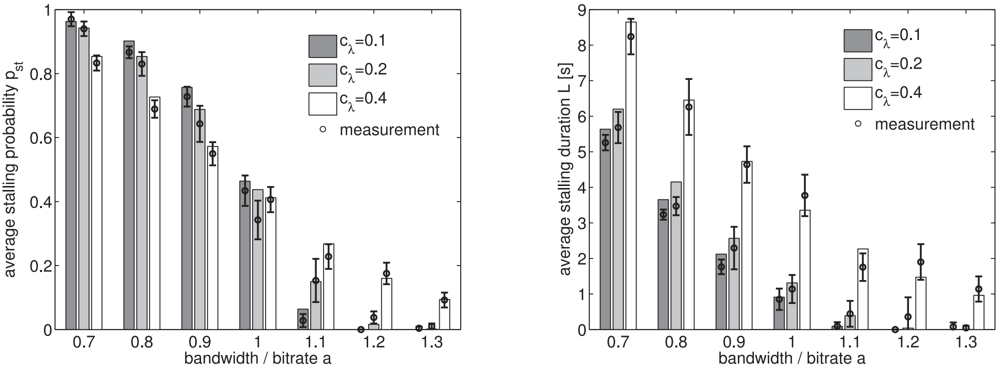
4.1 Internet Service Measurement

Figure 5 schematically depicts the testbed setup that is used to measure the number of stalling events at different link capacities. It consists of a network emulator NetEm (Hemminger 2005), a client, and a control PC. The client browses a video provided by a streaming server via the Internet. We use the control PC to manage the measurements to avoid interference with the experiments.

The NetEm machine is a SUN FIRE X4150 server with Ubuntu Server 14.04 LTS as the operating system, whereas the other PCs run Ubuntu 14.04 LTS. A Python script is used to start a packet capturing function on the client and to adjust the network configuration on NetEm. The script then activates Selenium WebDriver,[1] which is a tool that allows automating Web browsers. The tool is used to automatically browse the Web site for the video and start the download from the streaming server. We choose metacafe.com[2] as video streaming source, as it provides unencrypted adaptive video streaming.

We shape the bandwidth using NetEm according to the ratio of average video bitrate and available bandwidth, which is varied from 0.7 to 1.3. The video used for the measurements has a duration

[1] http://www.seleniumhq.org.
[2] http://www.metacafe.com.



(a) Stalling probability

(b) Average stalling duration

Fig. 6. Stalling probability and average stalling duration dependent on bandwidth and bandwidth variation.

of 508 s with a segment playtime of 10 s and an average video bitrate of 506kbit/s. To increase statistical significance of the measurements, we produce 20 replications for each bandwidth condition. The stalling events are captured on the application level by logging the player events.

We show the applicability of the model by comparing it to measurement results derived from experiments conducted in the testbed. In this case, the segment size is large enough so that the protocol overhead is negligible, meaning that the configured bandwidth can be used in the model, c.f., Assumption 3. The model parameters, i.e., the video playtime T, the segment playtime B_n, and the mean value and coefficient of variation (CoV) μ and c_μ of the segment bitrate distribution are set according to the video used in the measurement. The continue buffering threshold p is set to 30s, and the pause buffering threshold is set to 40s. Figure 6(a) and (b) show a comparison of the measurement results with the analytic model. The bars show the results derived with the analytical model. The markers show the mean value of 20 measurement repetitions, and the error bars depict 95% confidence intervals.

Figure 6(a) shows the stalling probability dependent on the bandwidth provisioning factor a for different CoVs of the bandwidth. The slight differences between the measurement and the analytic results can be explained by the fact that the bandwidth received in the testbed may not match exactly the configured bandwidth. Effects such as round-trip delays and protocol overheads may also have a slight impact on the measurements, which are not considered in the analytic model. Overall, the results confirm that the model is accurate, independent of the level and the variation of bandwidth. For low average bandwidth, a high bandwidth variation leads to fewer stalling events. This depends on the fact that the high bandwidth variation leads to higher peak bandwidths that allow downloading the segment in time. This effect reverses when the bandwidth level is higher than the bitrate such that $a > 1$. If the bandwidth level is constantly above the bitrate, each segment can be downloaded in time. Here, a high variation of the bandwidth can result in lacks of bandwidth that lead to stalling events.

Figure 6(b) shows the average stalling duration dependent on the bandwidth provisioning factor a for different CoVs of the bandwidth. Again, the results confirm that the model is accurate, independent of the level and the variation of bandwidth. The average stalling duration decreases with higher bandwidth. This depends on the fact that a higher bandwidth leads to a faster download of the next segment in case the video buffer runs empty. Independent of the bandwidth level, the average stalling duration increases with the bandwidth variation.

Table 2. Quality Layers of Sample Video

Layer	Resolution	Bitrate
180p	320x180	250kbps
270p	480x270	400kbps
360p	640x360	800kbps
540p	960x540	1.2Mbps
720p	1280x720	2.4Mbps
1080p	1920x1080	4.8Mbps

4.2 Controlled Artificial Environment

To diminish any external influences induced by, e.g., varying server or network load in the Internet service measurement, we set up a controlled artificial environment. The controlled artificial environment is set up on a virtual machine based on Ubuntu 14.04 LTS.

Open source sample streams are provided by bitmovin.com.[3] The videos are stored, including all quality layers, in either HLS or DASH format on the hard drive of the virtual machine. Table 2 shows the quality layers of the sample video. The sample video is an edited clip showing a sports competition containing slow-moving scenes, such as interviews and slow motions, and fast-moving scenes showing acrobatic moves. It has a duration of 210s and is segmented into segments of 4s. A simple Python server is set up to serve requests from a local host. A video player accesses the video stream by sending an HTTP-GET request to the M3U/MDP-file representing the HLS/DASH stream on a local host. To enable local traffic shaping, the traffic served from the local host is rerouted via the Intermediate Functional Block device (ifb).[4] The traffic is shaped on the ifb using NetEm to control the network condition during video buffering. This allows using practical network throughput traces in the measurements as well as varying the bandwidth according to a specific mean and standard deviation.

We use the open source player TAPAS (De Cicco et al. 2014), as it allows controlling the buffering parameters and logging all events necessary for the performance metrics. The modified version of the TAPAS player, which supports termination when the video stream ends and additional parameters, such as the initial delay threshold and the continue buffering threshold, is provided[5] in a fork of the repository. Stohr et al. (2017) show that the choice of the target buffer size together with the player implementation dominates the choice of the adaptation algorithms. As the target buffer size is the dominant factor on the playback quality, we configure TAPAS with the conventional rate adaptation algorithm (Li et al. 2014) as representative adaptation logic.

We use the controlled artificial environment to investigate the impact of the buffering threshold p on the stalling probability of the sample video. We configure the average bandwidth in 100kbps steps from 300kbps to 900kbps and calculate the effective bandwidth of each segment by the segment size and its download time. For increasing bandwidth, the bitrate generally also increases, as the adaptation logic downloads segments of higher layers, which have a higher bitrate. To account for the changing bitrate, we plot the stalling probability against the bandwidth provisioning factor a, i.e., the ratio of effective bandwidth and the average bitrate selected by the adaption logic. Figure 7 shows the results for a CoV of bandwidth per segment of 0.4. The mean values were derived from 30 measurement repetitions, and the error bars depict 95% confidence intervals. As in the Internet service measurement, the stalling probability decreases with the available bandwidth.

[3] https://bitmovin.com/mpeg-dash-hls-examples-sample-streams/.
[4] https://wiki.linuxfoundation.org/networking/ifb.
[5] https://github.com/v-burger/tapas.

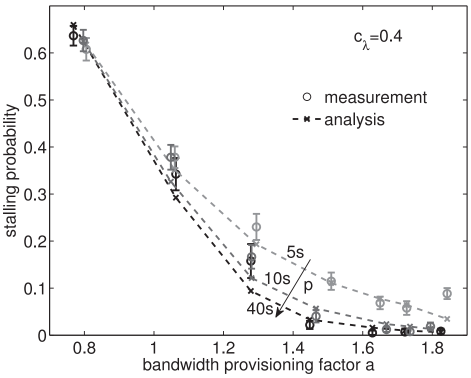


Fig. 7. Impact of threshold p on stalling probability of sample video.

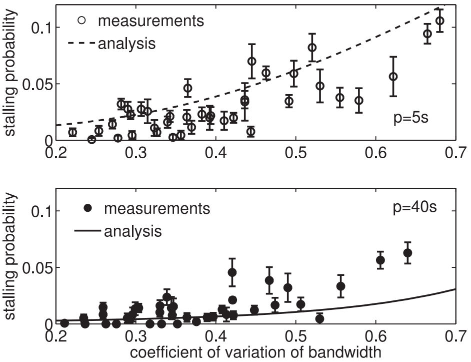


Fig. 8. Stalling probability of sample video in mobile throughput traces.

The effective bandwidth to bitrate ratio increases only slightly for high average bandwidths, as the adaptation logic increases the bitrate by selecting higher-quality layers. Hence, to increase the time on high-quality layers, the heuristic of HAS players tries to minimize the bandwidth provisioning factor a with the constraint $a > 1$ to prevent stalling events. This means that within a certain range for $a > 1$, the average bandwidth has low impact on the stalling probability because the adaptation logic keeps a low but high enough to prevent stalling events. The bandwidth provisioning factor a is only out of that range if the bandwidth is lower than the bitrate of the lowest-quality layer or substantially higher than the bitrate of the highest-quality layer.

By using a higher continue buffering threshold p, the average buffer level is higher if enough bandwidth is available, which means that the video stream is less likely to stall. This shows that the continue buffering threshold p has an influence on the buffer level and the stalling probability. Hence, the buffer policy can be optimized to maintain QoE and reduce traffic for adaptive video streams. Since the segment size in low-quality layers is small, the protocol overhead is not negligible in this case, c.f., Assumption 3. We calculate the stalling probability with the analytic model using the effective bandwidth and bandwidth variation received for downloading the segments in the measurements, as well as the average bitrate and the bitrate variation of the segments. The average bitrate and bitrate variation reflect the selection of quality layers by the adaptation logic. The model results are marked by an "x" and connected with dashed lines. Comparing the analytic results with the measurement results shows that the model well reflects the system dynamics for different buffering thresholds.

To show the impact of bandwidth variation and the buffer policy in real environments, we use bandwidth logs[6] derived from bandwidth measurements in 4G networks along several routes in and around the city of Ghent, Belgium, from December 16, 2015 to February 4, 2016. The bandwidth traces are measured traveling in different means of transport: foot, bicycle, bus, tram, train, or car. The videos are streamed starting at random points in time of the trace, and the traces were looped to guarantee termination of the stream. The measurement is repeated 30 times for each trace. To investigate if the available bandwidth or the bandwidth variation is a better indicator for the QoE of adaptive streams, we calculate the correlation with the average stalling probability over 30 runs. Table 3 shows the correlation coefficient with the stalling probability of the average bandwidth λ and the CoV of bandwidth c_λ received in each trace for different continue buffering

[6] http://users.ugent.be/~jvdrhoof/dataset-4g/.

Table 3. Correlation Coefficient With the Stalling Probability of the Average Bandwidth λ and the CoV of Bandwidth c_λ Received in Each Trace

p	Avg. Bandwidth λ	Bandwidth Variation c_λ	Analytic Model
5s	−0.60	0.79	0.92
10s	−0.52	0.74	0.97
40s	−0.47	0.72	0.98

thresholds p. The correlation coefficient is negative for the average bandwidth, as the stream stalls less frequently with increasing bandwidth. Comparing the absolute values of the average bandwidth and the bandwidth variation shows that the bandwidth variation is a better indicator and has more impact on the quality of an adaptive video stream than the average bandwidth. Furthermore, the correlation between average bandwidth and bandwidth variation decreases with increasing threshold p. This is due to the fact that with a higher threshold p, the dash heuristic starts to request new segments earlier, thus reducing the stalling probability. It should be noted that the correlation coefficient between the measured stalling probability and the corresponding stalling probability computed by the analytical model have a high correlation, as indicated by correlation coefficients > 0.92.

Figure 8 shows the mean stalling probability with 95% confidence intervals dependent on the bandwidth variation for a continue buffering threshold p of 5 and 40 s. The stalling probability is plotted dependent on the CoV of bandwidth per segment download in the trace. On average, the stalling probability is lower for a high buffering threshold showing the impact of the buffer policy on adaptive video streams in real environments. For increasing CoV of bandwidth, the stalling probability also tends to increase showing the impact of the bandwidth variation. The lines were derived by the analytic model showing the trend of the measurements results for increasing bandwidth variation. The analytic results were derived using the overall average bitrate and bitrate variation as well as the overall average bandwidth of the traces.

To enable repeatable and replicable measurements, the source code and the virtual machine configurations are published on GitHub.[7] The repository also contains a Vagrantfile, which allows automated creation and configuration of the controlled artificial environment. The Matlab implementation of the described analytical model is available on GitHub.[8]

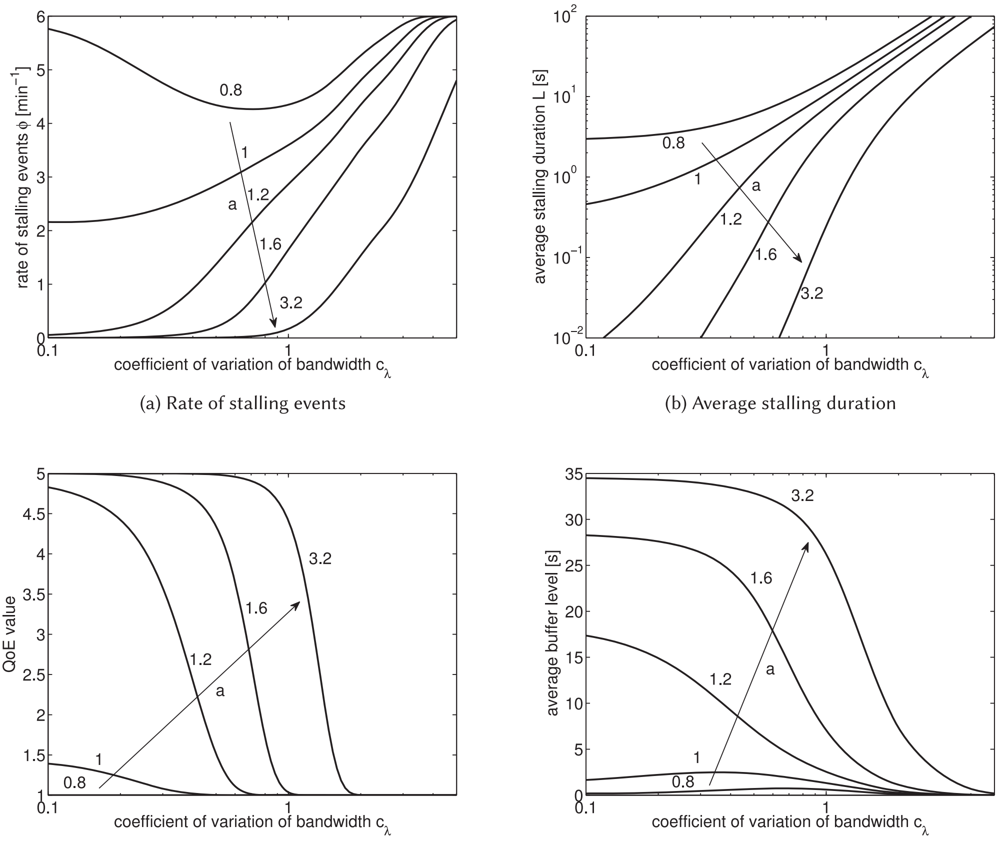
Our results show that the adaptation heuristic tries to keep a in a certain range above 1, which means that the average bandwidth has low impact on the stalling frequency of the video stream. We further show that the bandwidth variation is a better indicator for the quality of an adaptive video stream than the average bandwidth and that the continue buffering threshold p can be used to control the buffer level and reduce the stalling probability. We use our model to study the impact of the bandwidth variation and the buffer policy for different bandwidth conditions in the following.

5 NUMERICAL EXAMPLES

We evaluate the impact of bandwidth variation and the buffer policy on the quality of video streams in numerical examples. We first investigate the impact of the bandwidth variation and the model parameters on the playback quality based on QoE metrics. We then evaluate the impact of the setting of the buffering threshold and finally investigate the trade-off between the traffic wasted in case of video abortion and the QoE dependent on the buffer policy.

[7] https://github.com/v-burger/has-evalvm.

[8] https://github.com/v-burger/videobufferdta.



(a) Rate of stalling events

(b) Average stalling duration

(c) Quality of experience based on Q_1 model

(d) Average buffer level

Fig. 9. Rate of stalling events, average stalling duration, QoE value, and average buffer level dependent on the CoV of bandwidth c_λ.

5.1 Impact of Bandwidth Variation

The analytic model allows analyzing the impact of bandwidth variation on the QoE of video streams. Figure 9(a) shows the rate of stalling events dependent on the CoV of bandwidth c_λ in different bandwidth conditions. For a bandwidth provisioning factor greater than or equal to $a = 1.0$, the rate of stalling events increases with the CoV of bandwidth. This depends on the fact that the expected segment download time $E[A_n]$ increases with the variation of bandwidth, as the variance of the segment download throughput $VAR[D_n]$ is a factor in one term of the expected value of the ratio distribution, c.f., Equation (2). This fundamental relationship shows that proportionally more bandwidth is necessary to prevent stalling events for high bandwidth variation. For low bandwidth ($a = 0.8$), there is a minimum stalling rate for a CoV of $c_\lambda = 0.7$. For $c_\lambda < 0.7$, high bandwidth variation allows downloading segments in bandwidth bursts. If $c_\lambda > 0.7$, then this effect is outweighed by the increased expected segment download time induced by the variation of the bandwidth.

Figure 9(b) shows the duration of stalling events dependent on the CoV of bandwidth c_λ in different bandwidth conditions. The average stalling duration decreases with the available bandwidth. The bandwidth variation has a high impact on the average stalling duration. The stalling duration increases with the bandwidth variation, which depends on the fact that the expected segment download time increases with the bandwidth variation.

Based on the number and duration of stalling events, we can estimate the quality perceived by users, using the QoE-model defined in Equation (28). Figure 9(c) shows the QoE value dependent on the CoV of bandwidth c_λ in different bandwidth conditions. With increasing bandwidth, better QoE values are received, as expected. If the bandwidth is lower than the bitrate ($a = 0.8$), the QoE is bad, independent of the bandwidth variation, as the available bandwidth leads to a high frequency of stalling events with long average duration. If the bandwidth provisioning factor is greater than or equal to 1, then higher bandwidth variations lead to worse QoE values, as the rate and average duration of stalling events increases. A bandwidth provisioning factor of $a = 1$ can only work if the bandwidth is constant. However, due to variable bitrates, stalling events even occur for low bandwidth variations, resulting in QoE values of 1.5 or lower.

For $a = 1.2$, the bandwidth is sufficient to experience good streaming quality if the bandwidth variation is low. In this case, the bandwidth variation has a severe impact on the QoE as it drops from almost excellent quality (5) to bad quality (1), as the CoV of the bandwidth is increased from 0.1 to 0.7. The results of the measurements using mobile bandwidth traces in 3G and 4G networks (van der Hooft et al. 2016) show that in practice, the CoV of bandwidth per segment c_λ ranges from 0.2 to 0.7. This shows the relevance of the results, as especially the bandwidth variations measured in real networks can have a high impact on the QoE.

To evaluate the efficiency of the buffer policy in terms of traffic wasted in case of abortion of playback, we consider the average buffer level $\bar{u}_N$. Figure 9(d) shows the average buffer level dependent on the CoV of bandwidth c_λ for different bandwidth provisioning factors a. In case of a bandwidth provisioning factor less than or equal to 1, the average buffer level increases with higher bandwidth variation due to bandwidth bursts, if the bandwidth variation is low. If the bandwidth is higher than the bitrate, i.e., $a > 1$, the average buffer level decreases with the bandwidth variation. This depends on the fact that the buffer policy limits the maximum buffer level and that longer average segment download times caused by bandwidth variations reduce the average buffer level.

As the average buffer level and the QoE are both increasing on the buffering threshold p, the trade-off between a low average buffer level and a high QoE can be optimized by setting the thresholds appropriately.

5.2 Impact of Threshold Setting

To find optimal threshold settings that reduce the traffic wasted in case of video abortion and ensure high QoE values, we investigate the trade-off between the average buffer level $\bar{u}_N$ and the QoE dependent on the buffering thresholds. The difference between the pause buffering threshold q and the continue buffering threshold p determines the duration of the pause between two buffering periods. Hence, for a fixed p, the pause buffering threshold q mainly impacts the temporal pattern of the segment arrivals and has low impact on the average buffer level. Therefore, we set $q = p + 10\text{s}$ and only study the threshold p here.

Figure 10 shows the QoE dependent on the CoV of bandwidth c_λ for different thresholds p in different bandwidth conditions. Depending on the available bandwidth and the bandwidth variation, the threshold setting of p can have a high impact on the QoE. If the bandwidth is too low ($a \leq 1$), the quality cannot be increased by the buffer policy. For a bandwidth provisioning factor higher than 1, a certain QoE level can be kept by increasing p if the bandwidth variation increases. If the bandwidth provisioning factor is $a = 3.2$, a QoE of at least 4 is achieved with $p = 5\text{s}$ up to a bandwidth variation of $c_\lambda = 0.8$. If the bandwidth variation increases to $c_\lambda = 1$, the QoE would drop to 3. In this case, the QoE level can be kept at 4 if the buffer policy threshold is increased to $p = 20\text{s}$. This means especially in mobile environments where the bandwidth is highly varying,

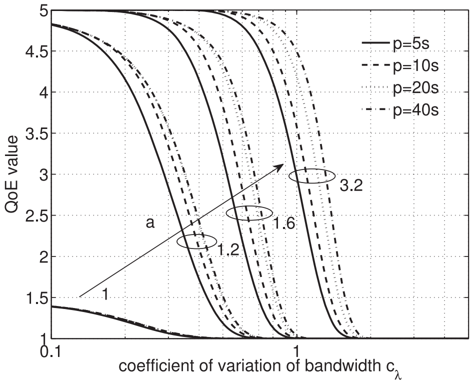


Fig. 10. QoE dependent on c_λ for different continue buffering thresholds p.

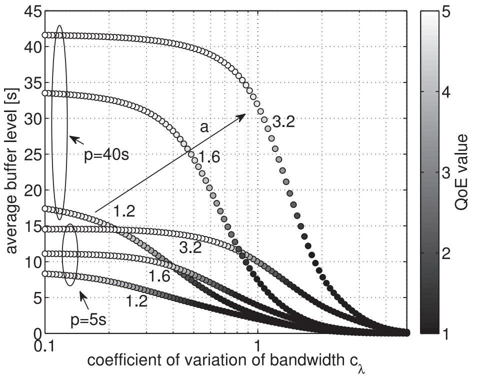


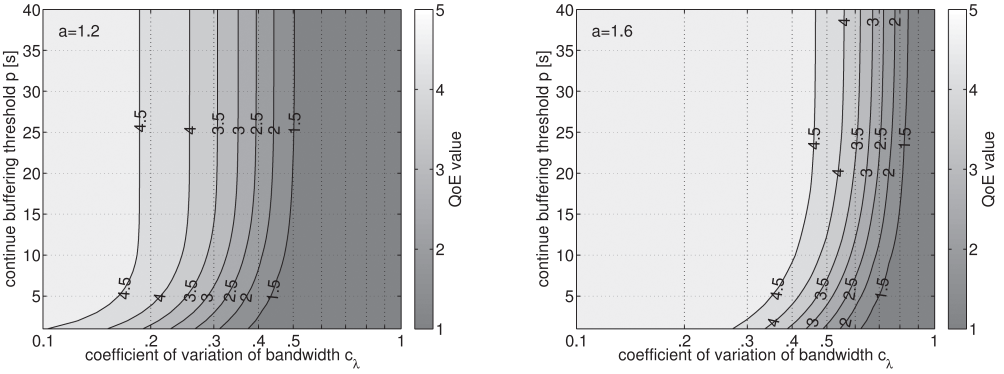
Fig. 11. Trade-off between QoE and average buffer level for different continue buffering thresholds p.

adapting the buffer policy thresholds to the bandwidth conditions has a high potential to improve video streaming services and the quality experienced by the users.

However, increasing p also increases the average buffer level and thus the traffic wasted in case of abortion. Therefore, it is important to consider the trade-off between QoE and average buffer level to determine in which cases an adaptation of p can improve the QoE and at what cost of potentially wasted traffic. Figure 11 shows the trade-off between QoE and average buffer level in different bandwidth conditions for a high and a low threshold p. The QoE value is coded in the color, ranging from black ($1 \equiv$ bad quality) to white ($5 \equiv$ excellent quality). High bandwidth variations cause longer average segment download times that reduce the average buffer level. A higher continue buffering threshold p leads to higher average buffer levels, which increases the traffic wasted in case of abortion, but also leads to fewer stalling events and thus leads to a better playback quality in terms of QoE. Dependent on the available bandwidth, the buffering threshold can be adapted to reduce the average buffer level while maintaining the desired QoE. If we consider a bandwidth provision factor of $a = 3.2$, a QoE value of 5 is achieved up to a bandwidth variation of $c_\lambda = 1$ using a buffering threshold of $p = 40$s. Up to a bandwidth variation of $c_\lambda = 0.7$, a QoE value of 5 is also achieved using a buffering threshold of $p = 5$s, which reduces the average buffer level and the traffic wasted in case of abortion to less than 15s, which is almost one third. Hence, to reduce the traffic wasted in case of abortion while maintaining perfect QoE, the buffering threshold can be set to $p = 5$s if $c_\lambda < 0.7$. If c_λ exceeds 0.7, p has to be increased such that a high QoE is maintained.

A video streaming service can set the buffering policy parameters to fulfill a certain targeted QoE value based on the network conditions. Figure 12 shows the threshold p necessary for a certain QoE value dependent on the bandwidth variation for two different network conditions with bandwidth provisioning factor $a = \{1.2, 1.6\}$. The QoE value, coded in color, ranges from grey ($1 \equiv$ bad quality) to white ($5 \equiv$ excellent quality). In the low bandwidth case (Figure 12(a)), a QoE value of at least 4 (good QoE or better) is only possible if the bandwidth variation per segment download is low with $c_\lambda < 0.2$. If $c_\lambda < 0.2$, the threshold p can be set to 5s to reduce the average buffer level, as the QoE is not improved for higher values of p. If the bandwidth variation exceeds $c_\lambda = 0.2$, p has to be increased to 10s or 15s to maintain a good QoE. Increasing p further does not improve the QoE. If $c_\lambda > 0.3$, the QoE values drop below 4, independent of p in this bandwidth condition.

If the bandwidth provisioning factor is slightly higher (Figure 12(b)), a QoE value of at least 4 is possible if $c_\lambda < 0.6$. Up to $c_\lambda = 0.2$, the QoE is perfect, independent of the threshold setting, and



(a) $a = 1.2$

(b) $a = 1.6$

Fig. 12. QoE dependent on the CoV of bandwidth c_λ for different thresholds p in different bandwidth conditions.

hence p can be set to a minimum value to reduce the average buffer level. To maintain perfect QoE, p has to be slightly increased if the bandwidth variation increases to $c_\lambda = 0.3$. The threshold p has to be further increased to 10s and 20s if c_λ increases to 0.4 and $c_\lambda = 0.5$, respectively. The QoE values drop for higher bandwidth variations, independent of p in this bandwidth condition.

6 CONCLUSION

In this study, we investigate the impact of network bandwidth and bitrate variations on the QoE of video streams dependent on the video buffer policy.

To this end, we model the video buffer as a GI/GI/1 queue with *pq*-policy using discrete-time analysis. In this context, the parameters p and q indicate thresholds where the video download is paused and resumed. This allows investigating the transient phase of the video buffer level, as well as its asymptotic behavior, by calculating the steady state of the buffer level distribution. In particular, the latter facilitates deriving theoretical bounds of the buffer level of adaptive video streaming services.

Results from measurements using an Internet streaming service and in a controlled artificial environment show that the bandwidth variation is a better indicator for the quality of an adaptive video stream than the average bandwidth and that the continue buffering threshold p can be used to control the buffer level and reduce the stalling probability. The results confirm that the model shows accurate results for adaptive video streams independent of the level and the variation of the bandwidth. Thus, we are able to study stochastic properties of the buffer-level distribution, including the probability for video stalls, based on network and video bitrate dynamics and estimate the corresponding impact on the user-perceived QoE.

Based on the derived model, we are able to quantify the negative impact of bandwidth variations on the QoE. Our results show that depending on the bandwidth variation, the buffering thresholds can have a high impact on the average buffer level and the stalling probability of adaptive video streams. It thus turns out that in cases with the same configuration and average bandwidth, already small bandwidth variations have a strong impact on the QoE. As a consequence, it is necessary to adjust the buffering parameters in such scenarios, particularly the threshold resuming the video download. In scenarios with moderate bandwidth variations, as typically perceived when driving with a bus or walking around in an urban environment, a good configuration for this threshold is 40s.

In parameter studies, we investigate the trade-off between average buffer level, i.e., costs for service providers in case of abortion of the video playback by the user, and QoE. Our results show that there is a high potential to optimize the parameters of *pq*-policy based on the specific network conditions. This can help Internet service and content providers by setting appropriate policy thresholds in each scenario and thus reduce the costs for video streaming while maintaining high QoE.

APPENDIX

A RATIO DISTRIBUTIONS

The ratio distribution $R = \frac{A}{B}$ of two uncorrelated normal distributed RVs $A \sim N(\mu_1, \sigma_1)$ and $B \sim N(\mu_2, \sigma_2)$ can be calculated according to Hinkley (1969).

For two generally distributed RVs C and D, and their ratio distribution $R = \frac{C}{D}$, we consider their probability mass functions $c(\kappa), d(\sigma)$, and $r(\tau)$ with domains $\kappa \in \{i_1, i_2, \ldots, i_m\}, \sigma \in \{j_1, j_2, \ldots, j_n\}$, and $\tau \in \{k_1, k_2, \ldots, k_o\}$ with $\sum_l c(l) = \sum_\sigma d(\sigma) = \sum_\tau r(\tau) = 1$. We calculate $r'(\frac{\kappa}{\sigma}) = c(\kappa) \cdot d(\sigma)$ and bin $\{\frac{\kappa}{\sigma} | \kappa \in \{i_1, i_2, \ldots, i_m\}, \sigma \in \{j_1, j_2, \ldots, j_n\}\}$ using the domain of the ratio distribution τ with equal intervals to derive a histogram of $r'(\frac{\kappa}{\sigma})$ and obtain the ratio distribution $r(\tau)$.

Both approaches have problems if the divisor has a high variance. In case of a normal distribution that is symmetric, the domain has negative values. The computational effort is too high to derive a high-precision histogram of the ratio distribution with generally distributed variables. Therefore, we use an approximation of the ratio distribution in our parameter studies. We approximate the target mean μ_t of the ratio distribution $R = \frac{A}{B}$ of two generally distributed RVs A and B by a Taylor expansion (Benaroya et al. 2005):

$$\mu_t = E\left[\frac{A}{B}\right] \approx \frac{E[A]}{E[B]} - \frac{COV[A,B]}{E[B]^2} + \frac{E[A]}{E[B]^3} VAR[B], \tag{30}$$

and the target standard deviation σ_t of the ratio distribution by

$$\begin{aligned} \sigma_t &= \sqrt{VAR\left[\frac{A}{B}\right]} = \sqrt{E\left[\left(A \cdot \frac{1}{B}\right)^2\right] - \left(E\left[A \cdot \frac{1}{B}\right]\right)^2} \\ &= \sqrt{E[A^2]E\left[\frac{1}{B^2}\right] - E[A]^2 E\left[\frac{1}{B}\right]^2}, \end{aligned} \tag{31}$$

where we derive $E[\frac{1}{B^2}]$ and $E[\frac{1}{B}]^2$ from a sampled distribution. We then approximate the ratio distribution by a log-normal distribution R with mean μ_t and standard deviation σ_t. To utilize R for discrete-time analysis, we truncate the distribution R with density $r(x)$ at the cap c and receive R_c with density $r_c(x)$ such that

$$r(x | X \le c) = \frac{r_c(x)}{R(c)}.$$

We finally readjust the mean value of R_c to fit μ_t in an iterative procedure.

REFERENCES

Velibor Adzic, Hari Kalva, and Borko Furht. 2011. Optimized adaptive HTTP streaming for mobile devices. In *SPIE Optical Engineering+Applications*. International Society for Optics and Photonics, 81350T–81350T.

K. R. Balachandran and Henk Tijms. 1975. On the D-policy for the M/G/1 queue. *Management Science* 21, 9, 1073–1076.

Wei Bao and Stefan Valentin. 2015. Bitrate adaptation for mobile video streaming based on buffer and channel state. In *Proceedings of the 2015 IEEE International Conference on Communications (ICC'15)*. IEEE, Los Alamitos, CA, 3076–3081.

Haym Benaroya, Seon Mi Han, and Mark Nagurka. 2005. *Probability Models in Engineering and Science*. Vol. 193. CRC Press, Boca Raton, FL.

W. Böhm and S. G. Mohanty. 1993. The transient solution of M/M/1 queues under (M, N)-policy. A combinatorial approach. *Journal of Statistical Planning and Inference* 34, 1, 22–33.

O. J. Boxma. 1976. Note—note on a control problem of Balachandran and Tijms. *Management Science* 22, 8, 916–917.

Cisco. 2017. *Cisco Visual Networking Index: Global Mobile Data Traffic Forecast Update, 2016-2021.* Technical Report. Cisco.

Luca De Cicco, Vito Caldaralo, Vittorio Palmisano, and Saverio Mascolo. 2014. TAPAS: A tool for rapid prototyping of adaptive streaming algorithms. In *Proceedings of the 2014 Workshop on Design, Quality, and Deployment of Adaptive Video Streaming.* ACM, New York, NY, 1–6.

Markus Fiedler, Tobias Hoßfeld, and Phuoc Tran-Gia. 2010. A generic quantitative relationship between quality of experience and quality of service. *IEEE Network* 24, 2, 36–41.

Stephen Hemmiger. 2005. Network emulation with NetEm. In *Proceedings of the Linux Conference.*

Daniel P. Heyman. 1977. The T-policy for the M/G/1 queue. *Management Science* 23, 7, 775–778.

D. V. Hinkley. 1969. On the ratio of two correlated normal random variables. *Biometrika* 56, 3, 635–639.

Tobias Hoßfeld, Sebastian Egger, Raimund Schatz, Markus Fiedler, Kathrin Masuch, and Charlott Lorentzen. 2012. Initial delay vs. interruptions: Between the devil and the deep blue sea. In *Proceedings of the 2012 4th International Workshop on Quality of Multimedia Experience (QoMEX'12).*

Tobias Hoßfeld, Christian Moldovan, and Christian Schwartz. 2015. To each according to his needs: Dimensioning video buffer for specific user profiles and behavior. In *Proceedings of the 2015 IFIP/IEEE International Symposium on Integrated Network Management (IM'15).*

Tobias Hoßfeld, Raimund Schatz, Ernst Biersack, and Louis Plissonneau. 2013a. Internet video delivery in YouTube: From traffic measurements to quality of experience. In *Data Traffic Monitoring and Analysis.* Springer.

Tobias Hoßfeld, Michael Seufert, Matthias Hirth, Thomas Zinner, Phuoc Tran-Gia, and Raimund Schatz. 2011. Quantification of YouTube QoE via crowdsourcing. In *Proceedings of the 2011 IEEE International Symposium on Multimedia (ISM'11).*

Tobias Hoßfeld, Michael Seufert, Christian Sieber, and Thomas Zinner. 2014. Assessing effect sizes of influence factors towards a QoE model for HTTP adaptive streaming. In *Proceedings of the 2014 6th International Workshop on Quality of Multimedia Experience (QoMEX'14).*

Tobias Hoßfeld, Lea Skorin-Kapov, Poul E. Heegaard, Martın Varela, and Kuan-Ta Chen. 2016. On additive and multiplicative QoS-QoE models for multiple QoS parameters. In *Proceedings of the 5th ISCA/DEGA Workshop on Perceptual Quality of Systems (PQS'16).*

Tobias Hoßfeld, Dominik Strohmeier, Alexander Raake, and Raimund Schatz. 2013b. Pippi Longstocking calculus for temporal stimuli pattern on YouTube QoE: 1+ 1= 3 and 1· 4≠ 4· 1. In *Proceedings of the 5th Workshop on Mobile Video.*

ITU-T RECOMMENDATION P.910. 1999. Subjective video quality assessment methods for multimedia applications. ITU-T.

C. James, E. Halepovic, M. Wang, R. Jana, and N. K. Shankaranarayanan. 2016. Is multipath TCP (MPTCP) beneficial for video streaming over DASH? In *Proceedings of the 2016 IEEE 24th International Symposium on Modeling, Analysis, and Simulation of Computer and Telecommunication Systems (MASCOTS'16).* IEEE, Los Alamitos, CA, 331–336.

Patrick Le Callet, Sebastian Möller, and Andrew Perkis. 2012. *Qualinet White Paper on dDefinitions of Quality of Experience.* European Network on Quality of Experience in Multimedia Systems and Services (COST Action IC 1003).

Zhi Li, Xiaoqing Zhu, Joshua Gahm, Rong Pan, Hao Hu, Ali C. Begen, and David Oran. 2014. Probe and adapt: Rate adaptation for HTTP video streaming at scale. *IEEE Journal on Selected Areas in Communications* 32, 4, 719–733.

Yao Liu, Sujit Dey, Fatih Ulupinar, Michael Luby, and Yinian Mao. 2015. Deriving and validating user experience model for DASH video streaming. *IEEE Transactions on Broadcasting* 61, 4, 651–665.

Tom H. Luan, Lin X. Cai, and Xuemin Shen. 2010. Impact of network dynamics on user's video quality: Analytical framework and QoS provision. *IEEE Transactions on Multimedia* 12, 1, 64–78.

Sami Mekki, Theodoros Karagkioules, and Stefan Valentin. 2017. HTTP adaptive streaming with indoors-outdoors detection in mobile networks. arXiv:1705.08809.

Christian Moldovan and Tobias Hoßfeld. 2016. Impact of variances on the QoE in video streaming. In *Proceedings of the 2016 28th International Teletraffic Congress (ITC-28).*

Juan J. Ramos-Muñoz, Jonathan Prados-Garzon, Pablo Ameigeiras, Jorge Navarro-Ortiz, and Juan M. López-Soler. 2014. Characteristics of mobile YouTube traffic. *IEEE Wireless Communications* 21, 18–25.

Michael Seufert, Sebastian Egger, Martin Slanina, Thomas Zinner, Tobias Hoßfeld, and Phuoc Tran-Gia. 2015. A survey on quality of experience of HTTP adaptive streaming. *IEEE Communications Surveys and Tutorials* 17, 1, 469–492.

Thomas Stockhammer. 2011. Dynamic adaptive streaming over HTTP: Standards and design principles. In *Proceedings of the 2nd Annual ACM Conference on Multimedia Systems.* ACM, New York, NY, 133–144.

Denny Stohr, Alexander Frömmgen, and Amr Rizk. 2017. Where are the sweet spots? A systematic approach to reproducible DASH player comparisons. In *Proceedings of the 2017 ACM Multimedia Conference (MM'17).* 1113–1121.

Phuoc Tran-Gia. 1993. Discrete-time analysis technique and application to usage parameter control modelling in ATM systems. In *Proceedings of the 8th Australian Teletraffic Research Seminar.*

J. van der Hooft, S. Petrangeli, T. Wauters, R. Huysegems, P. R. Alface, T. Bostoen, and F. De Turck. 2016. HTTP/2-based adaptive streaming of HEVC video over 4G/LTE networks. *IEEE Communications Letters* 20, 11, 2177–2180.

Florian Wamser, Pedro Casas, Michael Seufert, Christian Moldovan, Phuoc Tran-Gia, and Tobias Hoßfeld. 2016. Modeling the YouTube stack: From packets to quality of experience. *Computer Networks* 109, 211–224.

Xiufeng Xie, Xinyu Zhang, Swarun Kumar, and Li Erran Li. 2015. piStream: Physical layer informed adaptive video streaming over LTE. In *Proceedings of the 21st Annual International Conference on Mobile Computing and Networking (MobiCom'15)*. ACM, New York, NY, 413–425.

M. Yadin and P. Naor. 1967. On queueing systems with variable service capacities. *Naval Research Logistics Quarterly* 14, 1, 43–53.

Jian Yang, Han Hu, Hongsheng Xi, and Lajos Hanzo. 2011. Online buffer fullness estimation aided adaptive media playout for video streaming. *IEEE Transactions on Multimedia* 13, 5, 1141–1153.

Jian Yang, Yongyi Ran, Shuangwu Chen, Weiping Li, and Lajos Hanzo. 2016. Online source rate control for adaptive video streaming over HSPA and LTE-style variable bit rate downlink channels. *IEEE Transactions on Vehicular Technology* 65, 2, 643–657.

Runtong Zhang, Yannis Phillis, and Vassilis Kouikoglou. 2005. *Fuzzy Control of Queuing Systems*. Springer Science & Business Media.

Received June 2017; revised November 2017; accepted January 2018