

Automated Objective and Subjective Evaluation of HTTP Adaptive Streaming Systems

Christian Timmerer
Alpen-Adria-Universität Klagenfurt &
Bitmovin Inc.
Klagenfurt, Austria
christian.timmerer@itec.aau.at

Anatoliy Zabrovskiy
Petrozavodsk State University &
Alpen-Adria-Universität Klagenfurt
Petrozavodsk, Russia
z_anatoliy@petsu.ru

Ali C. Begen
Ozyegin University &
Networked Media, Inc.
Istanbul, Turkey
ali.begen@ozyegin.edu.tr

Abstract—Streaming audio and video content currently accounts for the majority of the internet traffic and is typically deployed over the top of the existing infrastructure. We are facing the challenge of a plethora of media players and adaptation algorithms showing different behavior but lack a common framework for both objective and subjective evaluation of such systems. This paper aims to close this gap by (i) proposing such a framework, (ii) describing its architecture, (iii) providing an example evaluation, (iv) and discussing open issues.

Keywords—HTTP Adaptive Streaming; objective evaluation; subjective evaluation; MPEG-DASH

I. INTRODUCTION

Adaptive bitrate (ABR) streaming or HTTP-based adaptive streaming (HAS) over the open, unmanaged internet has become the de-facto standard for the delivery of audiovisual content. Currently, the most popular two formats are (1) Dynamic Adaptive Streaming over HTTP (DASH) [1] and (2) HTTP Live Streaming (HLS) [2], which may converge to a single format in the near future thanks to the standardization of Common Media Application Format (CMAF) [3]. Independent of the representation format, the media is provided in multiple versions (*e.g.*, different resolutions and bitrates) and each version is divided into chunks of a few seconds (typically 2-10 s). A client first receives a manifest describing the available content on a server, and then, the client requests chunks based on its context (*e.g.*, observed available bandwidth, buffer status, and decoding capabilities). Thus, it is able to adapt the media presentation in a dynamic, adaptive way. In DASH, the chunks are referred to as segments and the manifest is called a media presentation description (MPD). In this paper, we use the terminology of DASH, however, this work can be also applied to any other format sharing the same principles.

In the past, we witnessed a plethora of research papers in this area, *i.e.*, surveys available here [4], [5], however, we still lack a comprehensive evaluation framework for HAS systems in terms of both objective – *i.e.*, Quality of Service (QoS) – and subjective – Quality of Experience (QoE) – metrics. Initial evaluations have been based on simple traffic shaping and network emulation tools [6] or means to rapidly

prototype the adaptation algorithms [7]. Recently, we have seen various evaluation frameworks in this domain focusing on adaptation algorithms proposed both in academia and industry [8]–[10]. However, the main focus has been on QoS rather than QoE. The latter typically requires user studies, which are mainly conducted within controlled laboratory environments. Yet, nowadays crowdsourcing is also considered as a reliable tool [11] and various platforms have been proposed [12] for this purpose.

In this paper, we propose a flexible and comprehensive framework to conduct objective and subjective evaluations of HAS systems in a fully automated and scalable way. It provides the following features:

- End-to-end HAS evaluation of players deployed in industry and algorithms proposed in academia under various conditions and use cases (*e.g.*, codecs/representations, network configurations, end user devices, player competition, *etc.*).
- Collection and analysis of objective streaming performance metrics (*e.g.*, startup time, stalls, quality switches, average bitrate).
- Subjective quality assessment utilizing crowdsourcing for QoE evaluation of HAS systems and QoE model testing/verification (*e.g.*, test or verify a proposed QoE model using subjective user studies).

The remainder of this paper is as follows. Section II comprises a detailed description of the architecture of the proposed framework. Section III presents example evaluation results to demonstrate the capabilities of the framework. A discussion and open research issues are provided in Section IV and Section V concludes the paper.

II. FRAMEWORK ARCHITECTURE

Our framework – as depicted in Figure 1 – supports both objective and subjective evaluation of HAS systems and is composed of *AdViSE* [13] and *WESP* [14] plus extensions.

AdViSE [13] ① is an adaptive video streaming evaluation framework for the automated testing of web-based media players and adaptation algorithms. It has been designed in an extensible way to support (i) different adaptive media

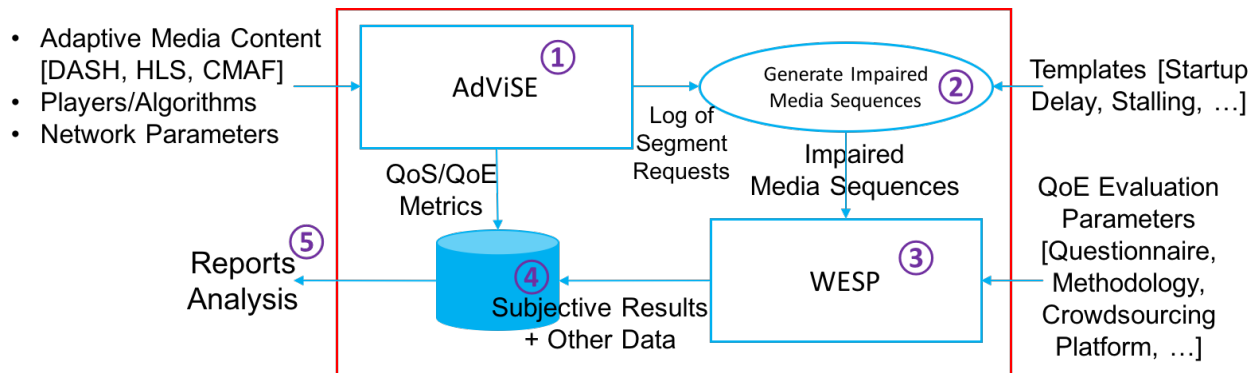


Figure 1: General framework architecture: **Adaptive Video Streaming Evaluation** framework for the automated testing of web-based media players and adaptation algorithms (**AdViSE**) & **Web-based Subjective evaluation Platform (WESP)**.

content formats (e.g., DASH, HLS, CMAF), (ii) commercially deployed media players as well as implementations of adaptation algorithms proposed in the research literature, and (iii) various networking parameters (e.g., bandwidth, delay) through network emulation. The output of AdViSE comprises a set of QoS and (objective) QoE metrics gathered and calculated during the adaptive streaming evaluation as well as a log of segment requests, which are used to generate the impaired media sequences ② used for the subjective evaluation.

The subjective evaluation is based on WESP [14] ③, which is a web-based subjective evaluation platform using existing crowdsourcing platforms for subject recruitment implementing best practices according to [11]. WESP takes the impaired media sequences as an input and allows for a flexible configuration of various QoE evaluation parameters such as (i) typical questionnaire assets (e.g., drop-down menus, radio buttons, free text fields), (ii) subjective quality assessment methodology based on ITU recommendations (e.g., absolute category rating), and (iii) different crowdsourcing platforms (e.g., Microworkers, Mechanical Turk). The output of WESP comprises the subjective results, e.g., mean opinion scores (MOS), and any other data gathered during the subjective quality assessment ④. Together with the outputs of AdViSE, it is used to generate fully automated reports and data export functions, which are eventually used for further analysis ⑤.

In the following we provide a detailed description of AdViSE and WESP focusing on how they connect with each other leading to a *fully automated objective and subjective evaluation of HAS systems*. Further details about the individual building blocks can be found in [8], [9], [13], [14].

A. AdViSE: Adaptive Video Streaming Evaluation

AdViSE includes the following components as depicted in Figure 2:

- Web server with standard HTTP hosting the media content and a MySQL database.

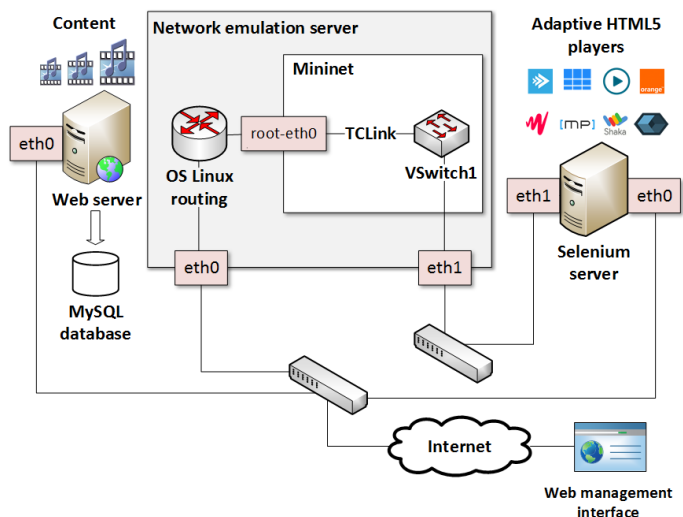


Figure 2: Architecture of adaptive video streaming evaluation framework for the automated testing of media players and adaptation algorithms.

- Network emulation server with a customized Mininet¹ environment for, e.g., bandwidth shaping.
- Selenium² servers for running adaptive media players/algorithms on various platforms. Note there might be multiple physical servers each hosting a limited set of players/algorithms.
- Web management interface for (i) conducting the experiments and (ii) running the adaptive media players.

AdViSE defines a flexible system that allows adding new adaptive media players/algorithms relatively fast. The *Web management interface* provides two functions, (i) for configuring and conducting the experiments, and (ii) including the actual player/algorithm to provide real-time information about the currently conducted experiment. Thus,

¹<http://mininet.org/>, accessed Feb. 20, 2018.

²<http://www.seleniumhq.org/>, accessed Feb. 20, 2018.

the proposed framework in this paper provides means for a comprehensive end-to-end evaluation of adaptive streaming services over HTTP including the possibility for subjective quality testing. The interface allows to define the following items and parameters: (1) configuration of network emulation profiles including the bandwidth trajectory, packet loss, and packet delay; (2) specification of the number of runs of an experiment; and (3) selection of one or more adaptive HTML5 player (or adaptation algorithm) and the adaptive streaming format used (*e.g.*, DASH, HLS, CMAF).

The result page provides a list of conducted experiments and the analytics section contains various metrics of the conducted experiments. It is possible to generate graphs for the results by using Highcharts³ and export the raw values for further offline analysis. The following quality parameters and metrics are currently available: (i) startup time; (ii) stalls (or buffer underruns); (iii) number of quality switches; (iv) download bitrate; (v) buffer length; (vi) average bitrate; (vii) instability and inefficiency; and (viii) simple QoE models specially designed for HAS. Further metrics can be easily added based on what the application programming interfaces (APIs) of players actually offer, and as new metrics or QoE models become available.

Finally, AdViSE provides the log of the segment requests, which are used – together with metrics such as startup time and stalls – to generate a media sequence as received by the player, and consequently, perceived by the user. The request log is used to concatenate the segments according to the request schedule of the player, thus, reflecting the media bitrate and quality switches. Other impairments such as startup time or stalls are automatically inserted based on the corresponding metrics gathered during the evaluation and by using predefined templates (*e.g.*, stalls displayed as spinning wheel). This impaired media sequence is used in the subsequent step for the subjective QoE evaluation using WESP, which could also include the unimpaired media presentation depending on the employed evaluation method.

Summary: AdViSE provides scalable, end-to-end HAS evaluation through emulation with a plethora of configuration possibilities regarding content configuration, players/algorithms (including for player competition), and network parameters. With AdViSE it is possible to utilize actual content and network settings with actual dynamic, adaptive streaming including rendering. We collect various metrics from players based on their API (*i.e.*, when access to source code is restricted) or from the algorithms/HTML5 directly. Additionally, we implemented so-called derived metrics and utilize QoE models proposed in the literature. Finally, the segment request log is used to generate impaired media sequence as perceived by end users for subjective quality testing.

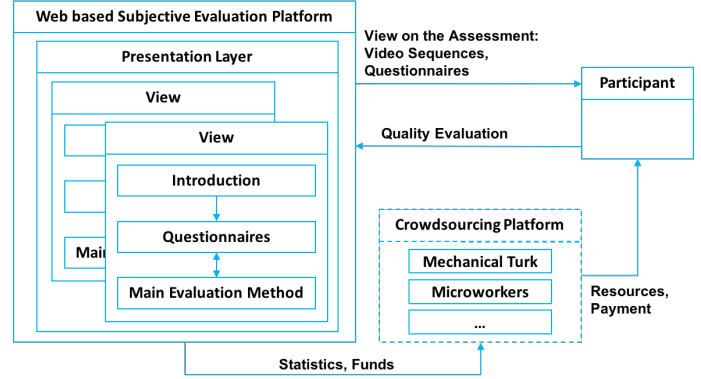


Figure 3: Architecture of web-based subjective evaluation platform.

B. WESP: Web-Based Subjective Evaluation Platform

Subjective quality assessments (SQAs) are used as a vital tool for evaluating QoE. SQAs provide reliable results, but is considered as cost-intensive and SQAs are typically conducted within controlled laboratory environments. Crowdsourcing has been proposed as an alternative to reduce the cost, however, various aspects need to be considered in order to get similar reliable results [11]. In the past, several frameworks have been proposed leveraging crowdsourcing platforms to conduct SQAs with each providing different features [12]. However, a common shortcoming of these frameworks is that they required tedious configuration and setup for each SQA, which made it difficult to use. Therefore, we propose to use a web-based management platform, which shall (i) enable easy and simple configuration of SQAs including possible integration of third-party tools for online surveys, (ii) provide means to conduct SQAs using the existing crowdsourcing platforms taking into account best practice as discussed in [11], and (iii) allow for the analysis of the results.

The goal of WESP is not only to provide a framework, which fulfills the recommendations of the ITU for subjective evaluations of multimedia applications (*e.g.*, BT.500⁴, P.910⁵, P.911⁶) but also provides the possibility to select and to configure the preferred evaluation method via a web interface. The conceptual WESP architecture, shown Figure 3, is implemented using HTML/PHP with MySQL database.

The introduction and questionnaires can be configured separately from the test methodology and may include control questions during the main evaluation. The voting possibility can be configured independently from the test methodology, providing more flexibility in selecting the appropriate voting mechanism and rating scale. The predefined

⁴<https://www.itu.int/rec/R-REC-BT.500>, accessed Feb. 20, 2018.

⁵<https://www.itu.int/rec/T-REC-P.910>, accessed Feb. 20, 2018.

⁶<https://www.itu.int/rec/T-REC-P.911>, accessed Feb. 20, 2018.

³<https://www.highcharts.com/>, accessed Feb. 20, 2018.

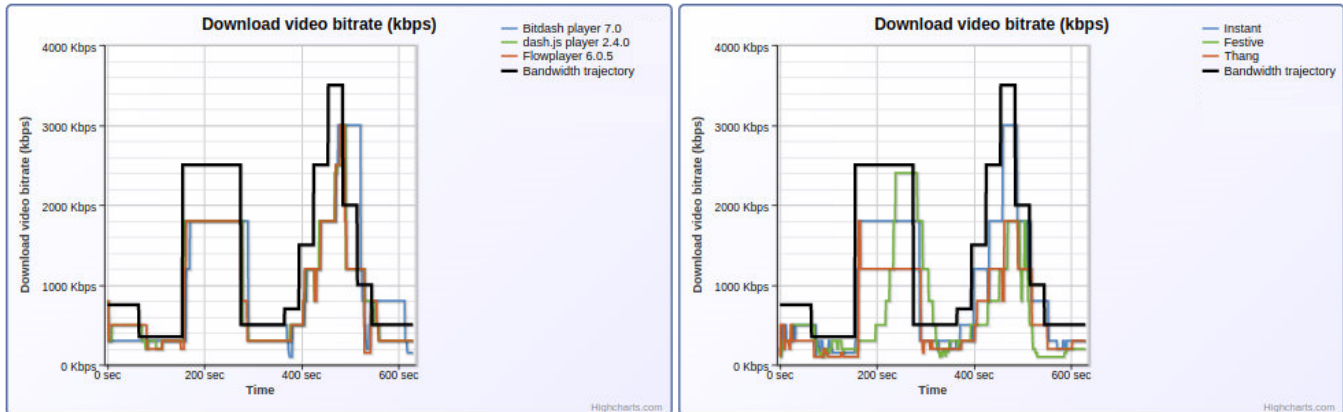


Figure 4: Download video bitrate for the selected industry players (left) and adaptation algorithms proposed in the research literature (right).

voting mechanisms include the common HTML interface elements and some custom controls like a slider in different variations. The platform consists of a management layer and a presentation layer. The management layer allows for maintaining the user study such as adding new questions or multimedia content and setting up the test method to be used (*e.g.*, single stimulus, double stimulus, pair comparison, continuous quality evaluation, *etc.*). The presentation layer is responsible for presenting the content to the participants. This allows providing different views on the user study, and thus, one can define groups to which the participants may be randomly (or in a predefined way) assigned. After a participant finishes the user study, the gathered data is stored in a MySQL database. Furthermore, the platform offers methods of tracking the participant’s behavior during a SQA (*e.g.*, focus of web browser’s window/tab, time for consuming each stimuli presentation, time it takes for the voting phase, *etc.*) and data provided by the web player API.

The stimuli presentation can be configured independently from the test method and may be combined with the voting possibility to support continuous quality evaluations. The media content can be fully downloaded and cached on the evaluation device prior starting the actual media presentation to avoid glitches during the evaluation, *e.g.*, due to network issues. However, it also supports streaming evaluation in real-world environments where various metrics (*e.g.*, startup time, stalls) are collected and stored for analysis.

Summary: WESP provides an extensible, web-based QoE evaluation platform utilizing crowdsourcing. It supports a plethora of evaluation methodologies and configuration possibilities. Although it has been specifically designed to implement SQAs for HAS systems using crowdsourcing (including support for real-world environments), it can be also used for SQAs within laboratory environments.

III. EXAMPLE EVALUATION RESULTS

In this section, we provide example evaluation results of selected industry players and adaptation algorithms proposed in the research literature: Bitmovin v7.0⁷, dash.js v2.4.0⁸, Flowplayer v6.0.5⁹, FESTIVE [15], Instant [16], Thang [17]. Note that we show only a small selection – also due to page limits – and the results presented here should be only seen as an example of what the framework provides rather than a full-fledged player comparison sheet. Additional further results using the tools described in this paper can be found in [8], [9], [18].

For the evaluation we used, the Big Buck Bunny sequence¹⁰ and encoded it according to the Amazon Prime video service, which offers 15 different representations as follows: 400x224 (100 Kbps), 400x224 (150), 512x288 (200), 512x288 (300), 512x288 (500), 640x360 (800), 704x396 (1200), 704x396 (1800), 720x404 (2400), 720x404 (2500), 960x540 (2995), 1280x720 (3000), 1280x720 (4500), 1920x1080 (8000), and 1920x1080 (15000). The segment length is 4 s and one audio representation at 128 Kbps is used. We adopt the bandwidth trajectory from [6] providing both step-wise and abrupt changes in the available bandwidth, *i.e.*, 750 Kbps (65 seconds), 350 Kbps (90), 2500 Kbps (120), 500 Kbps (90), 700 Kbps (30), 1500 Kbps (30), 2500 Kbps (30), 3500 Kbps (30), 2000 Kbps (30), 1000 Kbps (30) and 500 Kbps (85). The network delay is set to 70 milliseconds.

Figure 4 shows the download bitrate for the players and algorithms in question, and Table I provides an overview of all metrics. Metrics *a.–e.* are directly retrieved from the player/HTML5 API and algorithm implementation, respectively. Metrics *f.–g.* utilize simple QoE models [19], [20] to

⁷<https://bitmovin.com/>, accessed Feb. 20, 2018.

⁸<http://dashif.org/>, accessed Feb. 20, 2018.

⁹<https://flowplayer.com/>, accessed Feb. 20, 2018.

¹⁰<https://peach.blender.org/>, accessed Feb. 20, 2018.

Table I: Overview of example results.

	Bitmovin		dash.js		Flowplayer		FESTIVE		Instant		Thang	
	μ	σ	μ	σ	μ	σ	μ	σ	μ	σ	μ	σ
a. Startup time [s]	1.8	0.2	3.5	0.3	3.2	0.1	3.2	0.2	9.0	1.4	9.7	0.8
b. Stalls [#]	0	0	4	1,6	7	1,7	1	0,8	0	0	0	0
c. Stall duration [s]	0.0	0.0	5.4	3.4	14.2	3.5	1.0	1.0	0.0	0.0	0.0	0.0
d. Quality switches [#]	17	1	29	4	23	1	65	5	45	3	43	9
e. Bitrate [kbps]	982	22	891	9	905	10	664	10	916	19	617	14
f. QoE/MOS [Mäki] [19]	4.56	0.0	4.38	0.08	4.20	0.09	4.53	0.04	4.56	0.0	4.56	0.0
g. QoE/MOS [Mok] [20]	4.1	0.0	3.84	0.05	3.79	0.02	3.93	0.1	3.62	0.09	3.58	0.06

calculate MOS values ranging from one to five based on a subset of other metrics. Interestingly, industry players and research algorithms provide different performance behavior under the same conditions but can be directly compared among each other. A detailed analysis of the results is out of scope of this paper.

IV. DISCUSSION AND CHALLENGES

In this section, we provide a discussion about our framework for the automated objective and subjective evaluation of HAS systems. It allows for an easy setup of various configurations and running multiple evaluations in parallel. New players and algorithms can be added easily as they appear in the market and research literature. Over time it is possible to build up a repository of players and algorithms for comprehensive performance evaluation. As it is possible to run multiple Selenium servers in parallel, our framework is capable to evaluate when players/algorithms compete for bandwidth in various configurations (*e.g.*, n player A vs. m player B).

The framework is quite flexible, and thus, comes with a high number of degrees of freedom. Hence, it is important to design the evaluation carefully. Here we provide a brief list of the aspects to consider: (A) *Content assets*: content type, codec/coding parameters (including HDR, WCG), representations (bitrate/resolution pairs, also referred to as bitrate ladder), segment length (including GOP size), representation format (*i.e.*, DASH, HLS, CMAF), *etc.*; (B) *Network parameters*: bandwidth trajectory (*i.e.*, predefined, network traces), delay, loss, and other networking aspects (see below for further details); (C) *End user device environment*: device type, operating system, browser, *etc.*; (D) *Streaming performance metrics*: average bitrate, startup time, stalls (frequency, duration), quality switches (frequency, amplitude), *etc.*; (E) *Quantitative QoE models* based on audio-video quality and/or streaming performance metrics; (F) *General HAS evaluation setup*: live vs. on-demand content, single player vs. multiple players competing for bandwidth, *etc.*; (G) *Templates* for generating the impaired media sequence (*i.e.*, how to realize startup delay and stalls); (H) *Questionnaire* for SQA including *control questions* for crowdsourcing; (I) *SQA method* (*e.g.*, single

stimulus, double stimulus, pair-wise comparison) and its parametrization; and (J) Collection of all *results* and further (offline) *analysis*.

All these aspects are important to consider and a potential source of risk when conducting such experiments. A detailed analysis is beyond the scope of this paper.

Based on our experience when conducting multiple evaluations and performance comparisons, we identified the following research challenges, possibly subject to future work: (I) The reliability of results require cross-validation, specifically those from SQAs, which typically call for SQAs in controlled laboratory environments. (II) The network is a key aspect within HAS systems but often neglected. Network emulation is a vital tool but with limitations. For HAS systems, we also need to consider content distribution networks (CDNs), software-defined networking (SDN), information-centric networking (ICN), and next-generation (mobile) networks (*e.g.*, 5G). Detailed analysis and evaluations of these aspects in the context of HAS are currently missing. However, for example, recent standardization and research contributions showed benefits for HAS systems when combined with SDN [21]. (III) Reproducibility of such a framework can be achieved by providing containerized versions of the modules as done in [10]. This is considered critical when using industry players, which often require licenses. Additionally, it could be interesting to connect to large-scale research networks (such as PlanetLab¹¹, Virtual Internet Routing Lab¹², or GENI¹³).

V. CONCLUSIONS

This paper describes how AdViSE and WESP can be combined to perform objective and subjective evaluations of HAS systems in a fully automated and scalable way. For example, it can be used to test and compare new players/algorithms under various context conditions or research new QoE models with practically instant verification through subjective tests. The main finding of this work is, that a comprehensive objective and subjective evaluation of HAS systems is feasible for both industry players and adaptation

¹¹<https://www.planet-lab.org/>, accessed Feb. 20, 2018.

¹²<http://virl.cisco.com/getvirl/>, accessed Feb. 20, 2018.

¹³<http://www.geni.net/>, accessed Feb. 20, 2018.

algorithms proposed in the research literature. Hence, we recommend adopting it when proposing new features in this area and evaluating these features against the state of the art.

ACKNOWLEDGMENT

This work was supported in part by the Austrian Research Promotion Agency (FFG) under the Next Generation Video Streaming project “PROMETHEUS”.

REFERENCES

- [1] I. Sodagar, “The MPEG-DASH Standard for Multimedia Streaming Over the Internet,” *IEEE MultiMedia*, vol. 18, no. 4, pp. 62–67, Oct. 2011. [Online]. Available: <http://dx.doi.org/10.1109/MMUL.2011.71>
- [2] R. Pantos and W. May, “HTTP Live Streaming,” <https://www.ietf.org/rfc/rfc8216.txt>.
- [3] “Information technology – Multimedia application format (MPEG-A) – Part 19: Common media application format (CMAF) for segmented media,” ISO/IEC 23000-19, 2017.
- [4] M. Seufert, S. Egger, M. Slanina, T. Zinner, T. Hoßfeld, and P. Tran-Gia, “A Survey on Quality of Experience of HTTP Adaptive Streaming,” *IEEE Communications Surveys Tutorials*, vol. 17, no. 1, pp. 469–492, Firstquarter 2015.
- [5] Y. Sani, A. Mauthe, and C. Edwards, “Adaptive Bitrate Selection: A Survey,” *IEEE Communications Surveys Tutorials*, vol. 19, no. 4, pp. 2985–3014, Fourthquarter 2017.
- [6] C. Müller, S. Lederer, and C. Timmerer, “An Evaluation of Dynamic Adaptive Streaming over HTTP in Vehicular Environments,” in *Proceedings of the 4th Workshop on Mobile Video*, ser. MoVid ’12. New York, NY, USA: ACM, 2012, pp. 37–42. [Online]. Available: <http://doi.acm.org/10.1145/2151677.2151686>
- [7] L. De Cicco, V. Caldaralo, V. Palmisano, and S. Mascolo, “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*, ser. VideoNext ’14. New York, NY, USA: ACM, 2014, pp. 1–6. [Online]. Available: <http://doi.acm.org/10.1145/2676652.2676654>
- [8] A. Zabrovskiy, E. Petrov, E. Kuzmin, and C. Timmerer, “Evaluation of the Performance of Adaptive HTTP Streaming Systems,” *CoRR*, vol. abs/1710.02459, 2017. [Online]. Available: <http://arxiv.org/abs/1710.02459>
- [9] C. Timmerer, A. Zabrovskiy, E. Kuzmin, and E. Petrov, “Quality of Experience of Commercially Deployed Adaptive Media Players,” in *2017 21st Conference of Open Innovations Association (FRUCT)*, September 2017, pp. 330–335.
- [10] D. Stohr, A. Frömmgen, A. Rizk, M. Zink, R. Steinmetz, and W. Effelsberg, “Where Are the Sweet Spots?: A Systematic Approach to Reproducible DASH Player Comparisons,” in *Proceedings of the 2017 ACM on Multimedia Conference*, ser. MM ’17. New York, NY, USA: ACM, 2017, pp. 1113–1121. [Online]. Available: <http://doi.acm.org/10.1145/3123266.3123426>
- [11] T. Hoßfeld, C. Keimel, M. Hirth, B. Gardlo, J. Habigt, K. Diepold, and P. Tran-Gia, “Best Practices for QoE Crowdstesting: QoE Assessment With Crowdsourcing,” *IEEE Transactions on Multimedia*, vol. 16, no. 2, pp. 541–558, Feb 2014.
- [12] T. Hoßfeld, M. Hirth, P. Korshunov, P. Hanhart, B. Gardlo, C. Keimel, and C. Timmerer, “Survey of Web-based Crowdsourcing Frameworks for Subjective Quality Assessment,” in *2014 IEEE 16th International Workshop on Multimedia Signal Processing (MMSP)*, Sept 2014, pp. 1–6.
- [13] A. Zabrovskiy, E. Kuzmin, E. Petrov, C. Timmerer, and C. Mueller, “AdViSE: Adaptive Video Streaming Evaluation Framework for the Automated Testing of Media Players,” in *Proceedings of the 8th ACM on Multimedia Systems Conference*, ser. MMSys’17. New York, NY, USA: ACM, 2017, pp. 217–220. [Online]. Available: <http://doi.acm.org/10.1145/3083187.3083221>
- [14] B. Rainer, M. Walzl, and C. Timmerer, “A Web based Subjective Evaluation Platform,” in *2013 Fifth International Workshop on Quality of Multimedia Experience (QoMEX)*, July 2013, pp. 24–25.
- [15] J. Jiang, V. Sekar, and H. Zhang, “Improving Fairness, Efficiency, and Stability in HTTP-based Adaptive Video Streaming with FESTIVE,” in *Proceedings of the 8th International Conference on Emerging Networking Experiments and Technologies*, ser. CoNEXT ’12. New York, NY, USA: ACM, 2012, pp. 97–108. [Online]. Available: <http://doi.acm.org/10.1145/2413176.2413189>
- [16] L. Rubio Romero, “A Dynamic Adaptive HTTP Streaming Video Service for Google Android,” Master of Science Thesis, Royal Institute of Technology (KTH) Stockholm, Sweden, Oct 2011. [Online]. Available: <http://web.it.kth.se/~maguire/DEGREE-PROJECT-REPORTS/111006-Luciano-Rubio-with-cover.pdf>
- [17] T. C. Thang, Q.-D. Ho, J. W. Kang, and A. T. Pham, “Adaptive Streaming of Audiovisual Content using MPEG-DASH,” *IEEE Trans. Consumer Electronics*, no. 1, pp. 78–85.
- [18] C. Timmerer, M. Maiero, and B. Rainer, “Which Adaptation Logic? An Objective and Subjective Performance Evaluation of HTTP-based Adaptive Media Streaming Systems,” *CoRR*, vol. abs/1606.00341, 2016. [Online]. Available: <http://arxiv.org/abs/1606.00341>
- [19] T. Mäki, M. Varela, and D. Ammar, “A Layered Model for Quality Estimation of HTTP Video from QoS Measurements,” in *2015 11th International Conference on Signal-Image Technology Internet-Based Systems (SITIS)*, Nov 2015, pp. 591–598.
- [20] R. K. P. Mok, E. W. W. Chan, and R. K. C. Chang, “Measuring the Quality of Experience of HTTP Video Streaming,” in *12th IFIP/IEEE International Symposium on Integrated Network Management (IM 2011) and Workshops*, May 2011, pp. 485–492.
- [21] A. Bentaleb, A. C. Begen, R. Zimmermann, and S. Harous, “SDNHAS: An SDN-Enabled Architecture to Optimize QoE in HTTP Adaptive Streaming,” *IEEE Transactions on Multimedia*, vol. 19, no. 10, pp. 2136–2151, Oct 2017.