Electronic Communications of the EASST
Volume 080 (2021)

Conference on Networked Systems 2021 (NetSys 2021)

Towards QoE-Driven Optimization of Multi-Dimensional Content Streaming

Yassin Alkhalili, Jannis Weil, Anam Tahir, Tobias Meuser, Boris Koldehofe, Andreas Mauthe, Heinz Koeppl and Ralf Steinmetz

15 pages

Guest Editors: Andreas Blenk, Mathias Fischer, Stefan Fischer, Horst Hellbrueck, Oliver Hohlfeld, Andreas Kassler, Koojana Kuladinithi, Winfried Lamersdorf, Olaf Landsiedel, Andreas Timm-Giel, Alexey Vinel
ECEASST Home Page: http://www.easst.org/eceasst/ ISSN 1863-2122

Towards QoE-Driven Optimization of Multi-Dimensional Content Streaming

Yassin Alkhalili[1]*, Jannis Weil[1]*, Anam Tahir[2]*, Tobias Meuser[1], Boris Koldehofe[3], Andreas Mauthe[4], Heinz Koeppl[2] and Ralf Steinmetz[1]

[1] {yassin.alkhalili, jannis.weil, tobias.meuser, ralf.steinmetz}@kom.tu-darmstadt.de
Multimedia Communications Lab, Technical University of Darmstadt, Germany

[2] {anam.tahir, heinz.koeppl}@bcs.tu-darmstadt.de
Bioinspired Communication Systems Lab, Technical University of Darmstadt, Germany

[3] b.koldehofe@rug.nl
University of Groningen, the Netherlands

[4] mauthe@uni-koblenz.de
University of Koblenz-Landau, Germany

* These authors contributed equally

Abstract: Whereas adaptive video streaming for 2D video is well established and frequently used in streaming services, adaptation for emerging higher-dimensional content, such as point clouds, is still a research issue. Moreover, how to optimize resource usage in streaming services that support multiple content types of different dimensions and level of interactivity has so far not been sufficiently studied. Learning-based approaches aim to optimize the streaming experience according to user needs. They predict quality metrics and try to find system parameters maximizing them given the current network conditions. With this paper, we show how to approach content and network adaption driven by Quality of Experience (QoE) for multi-dimensional content. We describe components required to create a system adapting multiple streams of different content types simultaneously, identify research gaps and propose potential next steps.

Keywords: Multimedia Streaming, In-Network Processing, Reinforcement Learning, Quality of Experience

1 Introduction

Higher-dimensional content forms the basis of many multimedia applications, especially in the recently emerging domain of virtual and mixed reality [AGGS20]. Its use cases include telepresence with three-dimensional avatars and interactive objects, training and education in medicine, virtual property inspection, watching sports with a clear view, and playing video games. In the last few years, there has been an increased interest towards QoE enhancements of applications using higher-dimensional content [GPB+20], which adds additional parameters that need

to be considered. In this paper, we use the term Multi-Dimensional (MD) content encompassing lower-dimensional content types, like 2D video, and higher-dimensional content types, like point clouds. It has been that the advancement from conventional 2D video to higher-dimensional content, increases the required computational and network resources. At the same time, the level of user interactivity and immersiveness also increases [PTB+20]. To quantify a content type's level of interactivity, we use Degree of Freedom (DoF). DoF represents the number of independent variables a user can control to interact with the content. The 2D video has a DoF of 0 since it does not allow the users to interact with the content beyond changing the playback state. Users watching a 360-degree video can rotate their heads along three axes (3-DoF) and interaction with point clouds can additionally allow for movement in 3D space (6-DoF). Point clouds are a collection of points in the 3D space, each having a number of properties, including point coordinates along (X, Y, and Z) axes, color values encoded in RGB format, and several others. These points can be used to reconstruct a 3D object or an entire scene formed of numerous points. Point clouds can be captured using specialized cameras and depth sensors and can hold up to millions of points to depict high-quality reconstructed objects. Using these point clouds, the objects can be rendered from any viewing angle enabling a higher level of immersive viewing experience in comparison to 360-degree video. An increasing level of interactivity increases the difficulty of predicting which part of the content the users are going to consume next, thus reducing the performance of standard caching algorithms [PIT+18].

The transmission of higher-dimensional content through the network is much more resource-intensive than 2D content since the data volume increases manifold. Despite the increased capacity over the last years, the network resources are still limited and higher-dimensional content types put a much higher strain on them. Because of the direct interaction, the latency requirements are stricter for higher-dimensional content as well. For example, the authors in [RK15] show that latency greater than 20ms can lead to cybersickness in Virtual Reality (VR) applications. In the context of traditional 2D streaming, Quality of Service (QoS) is used to adapt the stream, by using measure network metrics such as jitter, latency, throughput, and packet loss rates. However, even for 2D content, it has been realized that the experience of the user cannot solely be captured by QoS parameters. QoE is a more user-oriented measure for the more subjectively perceived quality of a service [Int08]. There are many approaches that aim to improve the QoE of 2D video but finding a QoE metric which correctly reflects the requirements of higher-dimensional content types is still an open and challenging research area.

The concept of adaptation using mechanism transitions [AWB+19] can be leveraged for efficient streaming. The term *mechanism* refers to the algorithms used to process or transmit the content from the content providers to the end-user. Multiple mechanisms can fulfill the same purpose while using a different amount of network resources and may lead to different levels of QoE. Switching from one mechanism to another is called a *transition* and can be used to adapt to changing network conditions and user interactivity, in order to maintain the QoE for the user. For example, H.264 and HEVC/H.265 are two different types of video encoding mechanisms. H.265 compresses images more aggressively which reduces the bandwidth requirements but consequently, it requires almost ten times more computing power than H.264 [SM16]. Even though H.264 uses fewer resources as compared to H.265, links using H.264 could be overloaded by the much higher bandwidth requirements of 4k video, and thus lead to a decreased QoE for the user [UFŠV14]. Depending on the available network resources and the requirements of the content

type, transitions can be done between these two encoding mechanisms (and similarly amongst other available mechanisms) to improve the QoE of the current stream. However, in order to avoid having too much effect on the QoE of other parallel streams in the network, *cooperative transitions* are necessary. Cooperative transitions enable the entities to perform this adaptation collaboratively at multiple locations in the network, given all the available resources and requirements, with the aim that no stream is disadvantaged. Hence the aim is to develop a system that adapts to changing network conditions while maintaining the best possible QoE across competing media streams. This will be achieved through the use of cooperative transitions.

The main contributions of this paper are (i) the identification of the system architecture for QoE optimization of MD content distribution in the context of parallel (or competing) streams and (ii) the highlighting of the research gaps together with initial ideas on how to approach these.

The paper is structured as follows: In Section 2, we summarize related work regarding network and content adaptations with the aim of achieving a better QoS or QoE. In Section 3, we describe a general architecture for QoE optimization of MD content distribution, highlight the individual research gaps and propose first ideas. We then conclude this paper in Section 4.

2 Related work

The distribution of MD content is influenced by (i) network resources and conditions, (ii) the data path(s) from sender to receiver (flow), and (iii) the transmitted content itself. Flows are directly affected by the capabilities of the available resources of network nodes and end devices. In order to optimize the QoE, flows and their content can be adapted based on current network conditions, user preferences, and device characteristics. Current approaches can be categorized into network adaptation and content adaptation.

2.1 Adaptation of the network

Research on adaptation of the network mainly focuses on routing [SHL+19, XTM+18], scheduling [MSV+19] and congestion control [JRG+19] to optimize QoS metrics like average bandwidth, latency, packet loss and jitter. Leveraging Reinforcement Learning (RL) for network adaptation with respect to QoS is well researched [ZYK+09]. Many recent publications employ deep RL to adapt the network without requiring specific domain knowledge or handcrafted features and heuristics [JRG+19]. For example, Sun et al. use RL to create an Software-defined Networking (SDN) controller that improves the transmission delay in a real network environment by 9% as compared to traditional approaches [SHL+19]. Achieving a higher QoS at the receiver end is possible without changing the content and can also allow for a better QoE.

In addition, QoE can also directly be used as an optimization objective to improve the network's performance. As an example, Huang et al. propose to optimize network flows based on the end-to-end QoE of users consuming video streams [HYQR18]. They first create a Mean Opinion Score (MOS) model mapping selected QoS metrics at the receiver end to a MOS value, representing the QoE. The average predicted MOS is then used as a reward signal to learn a policy for a SDN controller with RL. They show that learning-based SDN traffic control can outperform selected baselines in terms of achieved QoE.

2.2 Adaptation of MD content

In the area of content adaptation, most publications focus on adaptive video streaming using adaptive bitrates (ABR). In Dynamic Adaptive Streaming over HTTP (DASH), a video is split into segments of fixed length, e.g. 5 seconds, that are encoded at different quality levels. The client fetches the next video segment for playback and determines a quality level based on the current network conditions. This way, the client can react to network impairments (e.g. a reduction in bandwidth) by lowering the video quality to keep a high QoE.

The system *Pensieve* [MNA17] leverages RL to construct algorithms for ABR video streaming. Given some QoS metrics representing the current network conditions, the main task of Pensieve is to predict the best bitrate for the next video segment with respect to the resulting QoE. It is trained using a simulation environment that aims to represent the basic dynamics of video streaming. The authors show that Pensieve is able to outperform state-of-the-art ABR schemes on a data set of over 30 hours of network traces and that it even generalizes to unseen networks. Using a similar approach, the system *Deeplive* extends this idea to ABR live video streaming and is able to outperform Pensieve in terms of QoE while having a lower computational overhead [TZN+19].

In the research study *Puffer* [YAZ+20], Yan et al. argue that QoE-based content adaptation using machine learning is hard because of the unknown network and user behavior. They show that huge testing datasets are required to differentiate between different methods with statistical significance. Additionally, their study reveals that the performance achieved in network simulations is not necessarily transferable to the real network. In particular, although Pensieve performs well in a simulated environment and was able to deliver good mean QoE values in a comparably small data set, Yan et al. show that it does not outperform a simple linear buffer-based control algorithm [HJM+14] when applied to the Internet. They propose a new control policy trained in a supervised manner on real traffic information and achieve better QoE than previous approaches.

In comparison to the 2D video, the QoE of 360-degree video and especially other interactive MD content types like point clouds is not extensively studied. However, quality adaptation schemes for these content types also exist [AMS20]. Most approaches used for 2D video streaming, like DASH, can also be applied to 360-degree video [KG18]. Instead of adapting the quality of a whole segment, it is possible to spatially divide the individual 360-degree video frames into smaller tiles [HTP+19, KRZ+19]. Fetching only the necessary tiles at adequate quality levels based on the user's orientation decreases the overall bandwidth requirements, which could allow for better QoE when combined with a good orientation prediction. Analogous to Pensive for 2D video, there exist approaches leveraging RL to adapt the content in 360-degree streaming applications [ZZB+19].

In the case of point cloud streaming, there are many publications that use meta-information about multiple versions of a scene with different quality levels to adaptively stream the content [HWD+19]. Extending the idea of DASH to point clouds, a client-side heuristic then chooses the most appropriate representation of individual objects or 3D tiles based on the network conditions, client device characteristics, and additional properties like the current viewport. As an example for viewport-based streaming of point clouds, [PCH18] splits point clouds into 3D tiles with a different Level of Detail (LoD) similar to the segmentation used in 360-degree video. Based on the user's view frustum, individual tiles of a scene can be either cut off completely or loaded

with a lower LoD. This can reduce the required bandwidth while maintaining the visual fidelity of the scene. Moreover, the authors introduce a rate-utility algorithm to distribute the available bandwidth amongst the tiles.

To reduce the resource requirements of higher-dimensional content types, the content can also be processed within the network. For example, Qian et al. [QHPG19] propose a point cloud streaming system for regular mobile devices. With edge computing, the point cloud stream is transcoded into a regular 2D pixel-based video that can then be efficiently transmitted and decoded by mobile devices, leveraging the existing advancements of adaptive 2D video streaming in terms of software and hardware acceleration. This system also includes QoE-based bitrate and viewport adaptation.

In contrast to network and 2D content adaptation, learning-based adaptation appears not to just yet be frequently used in applications featuring interactive higher-dimensional MD content.

3 System Architecture for MD Content Streaming

In this section, we propose a system architecture for MD content adaptation based on an example scenario and discuss the challenges and research gaps of its individual components in detail.

Figure 1 illustrates a scenario with three content providers distributing different types of MD content through the network to multiple users. The streams are represented as solid arrows. Higher-dimensional content allows users to interact with it, e.g. by turning their head. This is represented by a dotted line headed (toward the respective content provider). Flow may pass multiple *network domains*, which belong to different Internet providers and have certain network infrastructure (e.g. specific routers, switches, and servers). Each of these network domains employs different in-network processing mechanisms. For example, domain two contains a caching mechanism for 360-degree videos based on viewport prediction.

These mechanisms can also transform the flow, e.g. by filtering points of a 3D point cloud [ALRK19] or by rendering it as stereo 360-degree images. Such in-network processing mechanisms can enable users with low-end devices or weaker network connections to consume the content. However, they can also lead to quality degradation and higher delays in comparison to processing the MD content directly on the users' devices. Note that core network functions like routing can also be seen as mechanisms. In a realistic scenario, a content provider will only know its own network domains or the network domains of partners they cooperate with. This means there are also unknown domains that employ non-cooperative mechanisms. These unknown domains forward the MD content and affect the QoS of respective flows.

At the receiving node, the content is consumed by the users. Household 1 represents a mixed setting with 2D video and point cloud streaming, there is one mobile device streaming regular 2D video and household 2 contains one end device streaming 360-degree video. Users are connected to the Internet in various ways, e.g. via 5G, LAN, and Wi-Fi, and use different end devices, e.g. mobile phones, head-mounted devices, and a smart TV. Apart from the technical diversity, the users could also have different interaction patterns. For example, a person looking at a virtual exhibition in VR might interact with the content in a more predictable fashion than a person playing a fast-paced adventure game.

The QoE of the users depends on their end devices, the level of interactivity, the MD content

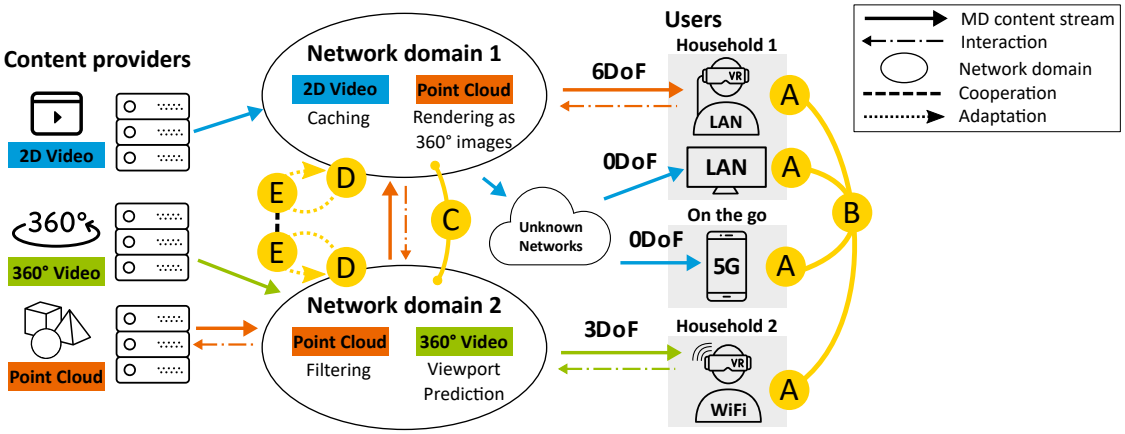

Figure 1: Three MD content types (left) are streamed over multiple network domains (center) to different end users (right). Our design compromises five main components to optimise the QoE: (A) assessment of the user's QoE, (B) aggregation of QoE considering fairness and guarantees, (C) overview of available mechanisms, (D) a system interface which allows to monitor and interact with the communication system, and (E) a policy to adapt the system's behaviour.

properties, the current network domain mechanisms, and the QoS. These properties are not entirely independent but rather influence each other, e.g. a different end device allows the user to interact with the content in a different way. The goal of each individual content provider is to achieve the highest possible QoE for their consumers.

An adaptive system should allow optimizing the QoE with transitions on network and content mechanisms while considering flows of multiple content types in parallel. We identify five main components of such a system, illustrated as yellow circles in Figure 1.

Firstly, it is infeasible to access the ground-truth QoE via user feedback at run-time. Hence, we need QoE models (A) to approximate user experience. As the system should be able to handle multiple MD content streams in parallel, the issue of fairness arises [HSHV17]. The available resources should be divided between all flows such that a fair QoE can be achieved. Therefore, a component (B) is needed to aggregate individual QoE values to a shared metric, accounting for user preferences and content types.

Adaptation is carried out by adapting currently applied mechanisms (e.g. changing the bitrate of a video stream) or by transitioning between functionally equivalent mechanisms (e.g. switching between different video encoding mechanisms with different performance and quality implications). This is only possible if there is a component (C) which allows querying the available transitions together with their effects and the requirements of underlying mechanisms.

In order to decide when adaptation is necessary, the system also needs to be aware of the current network conditions. Network monitoring is therefore required. Further, an abstract representation of the observable network state is necessary. This is facilitated through component (D). This information can then be leveraged by a policy (E) to predict the effect of potential adaptations under the current conditions. Using these estimates, individual communication systems can then be adapted independently or cooperatively to improve the QoE.

The following subsections will discuss each component Ⓐ to Ⓔ individually and suggest first steps on how to overcome corresponding challenges.

3.1 QoE Models for MD Content

Publications in the area of QoE often focus on traditional communication services and 2D video [BTB+19]. There are various authors which express QoE based on QoS metrics such as the Peak Signal to Noise Ratio (PSNR) [HWD+19]. In order to estimate the QoE without user studies, it is therefore common practice to create replacements for QoE models based on QoS [AW13, AM14]. However, to evaluate the actual user experience for MD content, user studies are inevitable. Measures like MOS and Likert Scales can then be used to quantify the QoE.

While a good understanding of QoE already exists for 2D content [SES+15], little is known about the quality factors of higher-dimensional content and their impact on the users [QHPG19]. In addition to traditional coding parameters such as resolution, colour depth, frame rate, affected regions, and quality variations, the QoE of MD content must also account for user interactivity and preferences [SKS+20]. Conducting user studies is the common approach to get direct feedback from the user to quantify QoE. The crucial part of the user studies is the design of the experiment and the questionnaire. In [HPP+17] and [KBHS20], the authors generated 360-degree videos that varied in the amount of motion they had, their perceptual quality, and the number of stalling events. They proposed questionnaires to measure the effect of the above variations on the QoE of the user in terms of perceptual quality, presence, acceptability, and cybersickness. MOS was used by user studies to answer questions such as, what is the threshold for the compression of videos, what is the impact of resolution while viewing with a head-mount display, is cybersickness an issue or not. However, the investigation of the impact of changing user interactivity on the QoE is still missing in these studies.

There is also a need for a quantitative formulation of the QoE in terms of these factors. Most of the research for higher-dimensional content is related to 360-degree video. Hence, there is a need to conduct user studies for 3D content like point clouds as well because we believe that the above-mentioned factors and conclusions would be different. To the best of our knowledge, the only case of QoE analysis for point clouds is by Sharabayko et. al. [SKS+20]. Here, the authors investigated the impact of two reduction mechanisms on the QoE of images that are generated from the reduced point clouds. Two kinds of audiences are involved in the study, which was conducted online with naive crowd workers besides point cloud experts. In a different study [HT18], the authors used a traditional image quality metric, namely PSNR, to assess the quality of point cloud objects under the effect of bit rate adaptation of the stream. However, as argued before, PSNR is not an appropriate replacement for accessing the actual user QoE.

3.2 QoE Fairness for MD Streams

The main goal of the adaptation is to improve the overall QoE with respect to the current network conditions. In the scope of this paper, the overall QoE is composed of the QoE of individual users consuming different MD content types. Each user should get the best experience possible, but the available resources should be distributed in a fair manner. Our notion of fairness refers to QoE fairness across users and MD content types. As the QoE metric should be comparable across MD

content types, this can be seen as a form of human-to-human fairness [BMA19]. In other words, multiple users consuming one content type (e.g. 2D video) should have a comparable QoE as users consuming other content types (e.g. point cloud streams). As content types have different resource requirements, this directly implies that there has to be traffic prioritization based on the content type and achievable QoE. Apart from the definition of fairness itself, we believe it will also be challenging to establish a QoE notion that is consistent and universally valid across all content types and end devices to facilitate comparisons among these metrics.

Current approaches considering multiple users often use simple aggregated metrics like the average QoE of all users to measure the overall quality of their adaptation [HYQR18]. Such metrics ignore fairness between users. For example, assume there are two users in the network and the QoE ranges from 1 (bad) to 5 (perfect). We can now describe the QoE of both users as a tuple (a, b) where $a, b \in [1, 5]$ is the QoE of user a and b respectively. The average QoE of $(1, 5)$ and $(3, 3)$ would be three in both cases. However, $(1, 5)$ can be considered unfair if $(3, 3)$ could be achieved using the available resources. We want to study how to optimize QoE distributions consisting of the QoE of multiple users while considering fairness.

Fairness is likely connected to the variance of the QoE across users. Intuitively, the optimal solution would be a configuration where the variance is zero and all users have the same QoE. In practice, optimality also depends on user preferences. Let's assume the content types of the previous example are (point cloud stream, 2D video). Point clouds are expected to have much higher resource requirements and the current network conditions might allow achieving both, $(3, 3)$ and $(2.5, 4.5)$. Although $(2.5, 4.5)$ can be considered unfair, a point cloud user might accept a minor drop in quality if this leads to big quality differences for other users. Especially if these users live in the same household. Therefore, QoE models can be augmented with utility metrics that account for user preferences. They can also be used to define QoE guarantees to avoid situations where too unevenly distributed resources lead to unacceptable levels of QoE for some users. If the QoE falls below a certain threshold, the utility drops to zero. The adaptive system can then distribute network resources based on these utility metrics to achieve a fair QoE.

3.3 Mechanism Profiles

To optimize the QoE based on the current network conditions, some aspects of the content distribution mechanisms have to be adapted. Possible adaptations can be categorized as follows:

- Exchanging mechanisms: A currently used mechanism is replaced with another functionally compatible mechanism (transition [AWB+19]), e.g., switching between different video encodings. See Sec. 1 for a detailed example.

- Reconfiguration of existing mechanisms: The underlying mechanism remains the same but certain properties are reconfigured (self-transition [Ric18]), e.g., updating routing information or changing the bitrate of a video stream to compensate for bandwidth changes.

- Adding and removing mechanisms: It is also possible to add and remove mechanisms in a specific flow. For example, a point cloud stream could be compressed between specific nodes. This can be transparent to the rest of the network.

In a communication system, there are multiple mechanisms running simultaneously and there can be mutual influence among different mechanisms. They can require different network resources and have a varying effect on the content stream and the QoE. Hence, there is a need to create mechanism profiles that describe the available mechanisms in a domain, the number of resources they need to give a certain output, and where they can be applied in a specific flow. These profiles should allow estimating the effects a mechanism has on QoS. This can then be used to estimate the QoE a mechanism can provide under certain network conditions. The adaptation process (E) will make use of these mechanism profiles to transition to suitable mechanisms to achieve a fair level of QoE (B) for all streams present.

3.4 Environment for Learning and Evaluation

Platforms used in network research can be categorized into real network environments, testbeds, network emulators, and network simulators. Each category has its advantages and disadvantages [HHJ+12]. Real network environments and emulators allow collecting information about real traffic while simulators typically allow for much faster data collection. However, simulators are based on abstract models of the network, thus creating a gap between the simulated and real network environments. Behavior learned in simulations might not be transferable to real networks. Unfortunately, learning-based approaches often require a huge amount of samples until the training process converges and the adaptive behavior generalizes over unseen samples. Solely using real network environments is therefore only possible for big content providers.

One of our goals is to create a meta-platform that leverages the advantages of the individual platforms. The core of this meta-platform will be an environment interface, similar to OpenAI Gym[1] but tailored for network experiments in the domain of MD content distribution. The interface should allow to include and combine specific implementations for our components (A) to (E) and therefore provide the foundation to support different MD content types. This also simplifies the comparison of different approaches as they then use the same main architecture.

This platform can then be connected to a network simulator like ns-3[2], OMNeT++[3] or Simonstrator [RSRS15], or used as a wrapper for real network environments. It would allow researchers working in any of these fields to include their systems and profit from potential synergies and easier comparability with existing solutions. Learning-based approaches can then be trained and evaluated in both environment types to close the gap between simulations and real networks. For example, an initial version might be trained only using simulators and its fine-tuning could be done in real networks. Using the standardized interface, it would also be possible to ground the simulation based on samples collected in real or emulated networks [HS17].

3.5 Adaptation of Parallel MD Streams

In reality, communication systems such as servers, routers, schedulers, and caches have limited access and can only carry out actions in their domains based on local observations. They are also limited in the set of actions they can perform. MD content distribution depends on the current

[1] See https://gym.openai.com/ (accessed 28.05.2021).
[2] See https://www.nsnam.org/ (accessed 28.05.2021).
[3] See https://omnetpp.org/ (accessed 28.05.2021).

network conditions and user interactivity. There is a need for the communication systems to adapt to different conditions to achieve a good QoE. This adaptation consists of transitions between available mechanisms while accounting for resource and content requirements. The network conditions of a single stream will also be affected by other streams sharing common limited resources (e.g. a home router). Therefore, it is desirable to consider cooperation across communication systems serving different content streams.

Our system architecture can naturally be seen as a RL problem. Each transition-capable element in the network is an agent which needs to learn how to adapt to the changing network and user conditions. All these agents will be connected to our environment interface can observe all parts of the network accessible or visible to them. For example, this could include the currently processed or forwarded traffic flows and available resources. Based on these observations, the network elements can decide which mechanisms to use for processing their flows and whether there is a need to transition between different mechanisms. These are the agent's actions. The goal of this adaptation is to successfully fulfill the QoE requirements of all the concurrent users in the network. Adaptation decisions must be taken quickly to meet time-critical QoE requirements. This means processing a huge amount of network data and extracting the important information from observations should be done efficiently and with sufficient accuracy. The reward of the system is an aggregated QoE metric considering fairness between multiple streams and QoE guarantees.

To solve the adaptation problem, we mainly want to focus on approaches in model-free RL [SB18] that do not require an explicit model of the environment. We expect the dimensionality of the observation and action spaces to be high, therefore approximate solution methods like deep neural networks can be used to represent the agent. Alternatively or in addition, the complexity and dimensionality of the adaptation problem itself could also be reduced with approaches such as variational autoencoders [KW14] and principal component analysis [DHB+13]. Such methods provide the agent with the key features which are the most influential on the effectiveness of the action, resulting in a faster and more refined decision-making process. Another thing to consider while adapting the system is the time and cost of the individual transitions. All actions should have a cost associated with them and do not necessarily show immediate effect after executing them. We believe that considering the temporal behavior of the adaptation could allow for a better transfer of learned behavior to real network environments. This will be especially relevant with respect to fulfilling time-critical QoE guarantees.

If each agent keeps optimizing the achievable QoE only in their local network domain, this can lead to local optima. To optimize its own flows, an agent might be selfish and overuse the resources of some other agent, leading to deterioration in the performance of the overall system. We believe that to achieve optimal QoE, these adaptations have to be performed over the entire data path of each flow. Achieving this in a centralized manner using one global entity can be computationally costly and does not scale well with an increasing number of flows and agents. Thus, a multi-agent system [Woo09] modeling is needed to decentralize this decision-making process and achieve better scalability. From the perspective of cooperative game theory [SL08], transition-capable communication systems could form coalitions with each other to achieve better QoE by performing cooperative transitions. These coalitions must take into account the interests of all communication systems, while several MD streams are processed by them in parallel.

Lastly, the network might also contain communication systems using the same limited re-

sources that are not capable of cooperation or transitions. Examples would be third-party video streams such as YouTube or Netflix. While they independently transition between quality levels based on the available bandwidth, they do not provide the functionality to explicitly cooperate with other streams. To perform cooperative transitions in their presence, behavioral estimates [Cam11] of these communication systems could help to better predict the network conditions for a reliable adaptation.

4 Conclusion

In this paper, we first provide a classification of the emerging content types and introduce the notion of MD content, comprising content from conventional 2D video to highly interactive 3D multimedia types. In the context of adaptive streaming, we categorize related work into two main categories: Adaptation of the network and adaptation of MD content. Instead of optimizing QoS metrics, content adaptation can also be performed with respect to an expected QoE. This allows taking into account the quality actually perceived by the users due to different characteristics of viewing and interacting with the content. While QoE-driven adaptation has been used for lower-dimensional content, modeling QoE of higher-dimensional content and the development of corresponding adaptive streaming techniques is still in its infancy.

Based on a scenario where multiple users interact with different MD content types, this paper describes an adaptive MD content streaming system based on five components. This system leverages cooperative transitions for a QoE-driven optimization of parallel MD content streams. Subsequently, the paper discusses the corresponding challenges and research gaps and provides first ideas on how they could be tackled in the near future. By establishing the bigger picture of adaptive MD content streaming, we believe that this paper will facilitate new collaborations and inspire other researchers working in this field.

Acknowledgements: This work has been co-funded by the German Research Foundation (DFG) as part of the projects C3 and B1 in the Collaborative Research Center (CRC) 1053 MAKI.

Bibliography

[AGGS20] K. Ahir, K. Govani, R. Gajera, M. Shah. Application on Virtual Reality for Enhanced Education Learning, Military Training and Sports. *Augmented Human Research* 5(1):1–9, 2020.

[ALRK19] Y. Alkhalili, M. Luthra, A. Rizk, B. Koldehofe. 3-D Urban Objects Detection and Classification From Point Clouds. DEBS '19. ACM, 2019.

[AM14] S. Aroussi, A. Mellouk. Survey on Machine Learning-based QoE-QoS Correlation Models. In *2014 International Conference on Computing, Management and Telecommunications (ComManTel)*. Pp. 200–204. 2014.

[AMS20] Y. Alkhalili, T. Meuser, R. Steinmetz. A Survey of Volumetric Content Streaming Approaches. In *2020 IEEE Sixth International Conference on Multimedia Big Data (BigMM)*. Pp. 191–199. 2020.

[AW13] M. Alreshoodi, J. Woods. Survey on QoE\QoS Correlation Models for Multimedia Services. *arXiv preprint arXiv:1306.0221*, 2013.

[AWB+19] B. Alt, M. Weckesser, C. Becker, M. Hollick, S. Kar, A. Klein, R. Klose, R. Kluge, H. Koeppl, B. Koldehofe et al. Transitions: A Protocol-Independent View of the Future Internet. *Proceedings of the IEEE* 107(4):835–846, 2019.

[BMA19] A. O. Basil, M. Mu, A. Al-Sherbaz. A Software Defined Network based research on Fairness in Multimedia. In *Proceedings of the 1st International Workshop on Fairness, Accountability, and Transparency in MultiMedia*. Pp. 11–18. 2019.

[BTB+19] A. Bentaleb, B. Taani, A. C. Begen, C. Timmerer, R. Zimmermann. A Survey on Bitrate Adaptation Schemes for Streaming Media Over HTTP. *IEEE Communications Surveys Tutorials* 21(1):562–585, 2019.

[Cam11] C. F. Camerer. *Behavioral Game Theory: Experiments in Strategic Interaction*. Princeton University Press, 2011.

[DHB+13] U. Demšar, P. Harris, C. Brunsdon, A. S. Fotheringham, S. McLoone. Principal Component Analysis on Spatial Data: An Overview. *Annals of the Association of American Geographers* 103(1):106–128, 2013.

[GPB+20] S. Gül, D. Podborski, T. Buchholz, T. Schierl, C. Hellge. Low-Latency Cloud-based Volumetric video Streaming using Head Motion Prediction. In *Proceedings of the 30th ACM Workshop on Network and Operating Systems Support for Digital Audio and Video*. Pp. 27–33. 2020.

[HHJ+12] N. Handigol, B. Heller, V. Jeyakumar, B. Lantz, N. McKeown. Reproducible Network Experiments using Container-based Emulation. In *Proceedings of the 8th international conference on Emerging networking experiments and technologies*. Pp. 253–264. 2012.

[HJM+14] T.-Y. Huang, R. Johari, N. McKeown, M. Trunnell, M. Watson. A Buffer-based Approach to Rate Adaptation: Evidence from a large video Streaming Service. In *Proceedings of the 2014 ACM conference on SIGCOMM*. Pp. 187–198. 2014.

[HPP+17] T. Huyen, N. Pham Ngoc, C. Pham, Y. Jung, T. Cong Thang. A Subjective Study on QoE of 360 video for VR Communication. In *19th IEEE Int. Workshop on Multimedia Signal Processing (MMSP)*. Pp. 1–6. 2017.

[HS17] J. Hanna, P. Stone. Grounded Action Transformation for Robot Learning in Simulation. In *Proceedings of the AAAI Conference on Artificial Intelligence*. Volume 31(1). 2017.

[HSHV17] T. Hoßfeld, L. Skorin-Kapov, P. E. Heegaard, M. Varela. Definition of QoE Fairness in Shared Systems. *IEEE Communications Letters* 21(1):184–187, 2017.

[HT18] M. Hosseini, C. Timmerer. Dynamic Adaptive Point Cloud Streaming. In *23rd Packet Video Workshop (PV)*. Pp. 25–30. 2018.

[HTP+19] J. van der Hooft, M. Torres Vega, S. Petrangeli, T. Wauters, F. De Turck. Tile-based adaptive streaming for virtual reality video. *ACM Transactions On Multimedia Computing Communications And Applications* 15(4):24, 2019.

[HWD+19] J. van der Hooft, T. Wauters, F. De Turck, C. Timmerer, H. Hellwagner. Towards 6DoF HTTP Adaptive Streaming through Point Cloud Compression. In *27th ACM Int. Conference on Multimedia (MM)*. Pp. 2405–2413. 2019.

[HYQR18] X. Huang, T. Yuan, G. Qiao, Y. Ren. Deep Reinforcement Learning for Multimedia Traffic Control in Software Defined Networking. *IEEE Network* 32(6):35–41, 2018.

[Int08] International Telecommunication Union (ITU). Quality of experience requirements for IPTV services. ITU-T Rec. G.1080, 2008.

[JRG+19] N. Jay, N. H. Rotman, B. Godfrey, M. Schapira, A. Tamar. A Deep Reinforcement Learning Perspective on Internet Congestion Control. In *Proceedings of the 36th International Conference on Machine Learning (ICML)*. Volume 97, pp. 3050–3059. 2019.

[KBHS20] A. van Kasteren, K. Brunnström, J. Hedlund, C. Snijders. Quality of Experience Assessment of 360-degree video. *Electronic Imaging* 2020(11):91–1, 2020.

[KG18] K. Khan, W. Goodridge. Future DASH Applications: A Survey. *Int. Journal of Advanced Networking and Applications* 10(2):3758–3764, 2018.

[KRZ+19] C. Koch, A.-T. Rak, M. Zink, R. Steinmetz, A. Rizk. Transitions of Viewport Quality Adaptation Mechanisms in 360 Degree Video Streaming. In *29th ACM Workshop on Network and Operating Systems Support for Digital Audio and Video (NOSSDAV)*. Pp. 14–19. 2019.

[KW14] D. P. Kingma, M. Welling. Auto-Encoding Variational Bayes. In *2nd International Conference on Learning Representations (ICLR)*. 2014.

[MNA17] H. Mao, R. Netravali, M. Alizadeh. Neural Adaptive Video Streaming with Pensieve. In *Conference of the ACM Special Interest Group on Data Communication (SIGCOMM)*. Pp. 197–210. 2017.

[MSV+19] H. Mao, M. Schwarzkopf, S. B. Venkatakrishnan, Z. Meng, M. Alizadeh. Learning Scheduling Algorithms for Data Processing Clusters. In *Conference of the ACM Special Interest Group on Data Communication (SIGCOMM)*. Pp. 270–288. 2019.

[PCH18] J. Park, P. A. Chou, J. Hwang. Volumetric Media Streaming for Augmented Reality. In *IEEE Global Communications Conference (GLOBECOM)*. Pp. 1–6. 2018.

[PIT+18] G. S. Paschos, G. Iosifidis, M. Tao, D. Towsley, G. Caire. The role of caching in future communication systems and networks. *IEEE Journal on Selected Areas in Communications* 36(6):1111–1125, 2018.

[PTB+20] A. Perkis, C. Timmerer, S. Baraković, J. B. Husić, S. Bech, S. Bosse, J. Botev, K. Brunnström, L. Cruz, K. De Moor et al. QUALINET white paper on definitions of immersive media experience (IMEx). *arXiv:2007.07032*, 2020.

[QHPG19] F. Qian, B. Han, J. Pair, V. Gopalakrishnan. Toward Practical Volumetric Video Streaming on Commodity Smartphones. In *20th Int. Workshop on Mobile Computing Systems and Applications (HotMobile)*. Pp. 135–140. 2019.

[Ric18] B. Richerzhagen. *Mechanism Transitions in Publish/Subscribe Systems: Adaptive Event Brokering for Location-based Mobile Social Applications*. Springer, 2018.

[RK15] K. Raaen, I. Kjellmo. Measuring Latency in Virtual Reality Systems. In Chorianopoulos et al. (eds.), *Proc. 14th International Conference on Entertainment Computing (ICEC)*. Lecture Notes in Computer Science 9353, pp. 457–462. Springer, 2015.

[RSRS15] B. Richerzhagen, D. Stingl, J. Rückert, R. Steinmetz. Simonstrator: simulation and prototyping platform for distributed mobile applications. In Theodoropoulos (ed.), *Proceedings of the 8th International Conference on Simulation Tools and Techniques, Athens, Greece, August 24-26, 2015*. Pp. 99–108. ICST/ACM, 2015.

[SB18] R. S. Sutton, A. G. Barto. *Reinforcement learning: An introduction*. MIT press, 2018.

[SES+15] M. Seufert, S. Egger-Lampl, M. Slanina, T. Zinner, T. Hossfeld, P. Tran-Gia. A Survey on Quality of Experience of HTTP Adaptive Streaming. *IEEE Communications Surveys Tutorials* 17(1):469–492, 2015.

[SHL+19] P. Sun, Y. Hu, J. Lan, L. Tian, M. Chen. TIDE: Time-relevant deep reinforcement learning for routing optimization. *Future Gener. Comput. Syst.* 99:401–409, 2019.

[SKS+20] M. Seufert, J. Kargl, J. Schauer, A. Nuchter, T. Hossfeld. Different Points of View: Impact of 3D Point Cloud Reduction on QoE of Rendered Images. In *12th International Conference on Quality of Multimedia Experience (QoMEX)*. Pp. 1–6. 2020.

[SL08] Y. Shoham, K. Leyton-Brown. *Multiagent systems: Algorithmic, game-theoretic, and logical foundations*. Cambridge University Press, 2008.

[SM16] M. Sharabayko, N. Markov. Contemporary video compression standards: H. 265/HEVC, VP9, VP10, Daala. In *2016 International Siberian Conference on Control and Communications (SIBCON)*. Pp. 1–4. 2016.

[TZN+19] Z. Tian, L. Zhao, L. Nie, P. Chen, S. Chen. Deeplive: QoE Optimization for Live video Streaming through Deep Reinforcement Learning. In *2019 IEEE 25th International Conference on Parallel and Distributed Systems (ICPADS)*. Pp. 827–831. 2019.

[UFŠV14] M. Uhrina, J. Frnda, L. Ševčík, M. Vaculik. Impact of H. 264/AVC and H. 265/HEVC Compression Standards on the video Quality for 4K Resolution. 2014.

[Woo09] M. Wooldridge. *An Introduction To Multiagent Systems*. John Wiley & Sons, 2009.

[XTM+18] Z. Xu, J. Tang, J. Meng, W. Zhang, Y. Wang, C. H. Liu, D. Yang. Experience-Driven Networking: A Deep Reinforcement Learning based Approach. In *IEEE Conference on Computer Communications (INFOCOM)*. Pp. 1871–1879. 2018.

[YAZ+20] F. Y. Yan, H. Ayers, C. Zhu, S. Fouladi, J. Hong, K. Zhang, P. Levis, K. Winstein. Learning in situ: a randomized experiment in video streaming. In *17th USENIX Symposium on Networked Systems Design and Implementation (NSDI 20)*. Pp. 495–511. 2020.

[ZYK+09] Y. Zhou, M. Yun, T. Kim, A. Arora, H.-A. Choi. RL-Based Queue Management for QoS Support in Multi-Channel Multi-Radio Mesh Networks. In *8th IEEE Int. Symposium on Network Computing and Applications (NCA)*. Pp. 306–309. 2009.

[ZZB+19] Y. Zhang, P. Zhao, K. Bian, Y. Liu, L. Song, X. Li. DRL360: 360-degree Video Streaming with Deep Reinforcement Learning. In *IEEE Conference on Computer Communications (INFOCOM)*. Pp. 1252–1260. 2019.