Summary

Computer networks are shared by multiple users and devices. Traffic from multiple applications is combined on a single network link, dividing the bandwidth between these applications. Video streaming applications require a relatively large and stable share of the bandwidth to be able to deliver high quality video. The architecture and protocols used by the current dominant streaming technology, HTTP Adaptive Streaming (HAS), have enabled the massive scale of video content delivery as we know it today. Yet they are not suitable for meeting the demanding bandwidth requirements in shared networks. Cross-traffic from other applications and other video streams have a large impact, causing HAS players to lower the video quality, frequently undershoot and overshoot the video quality compared to the available bandwidth, and even interrupt the stream for rebuffering.

Failure to deliver high quality video streams reflects badly on content providers and network operators, who risk the loss of revenue due to unsatisfied users. Video delivery problems distract the user, cause annoyance, and may lead to users abandoning their streams. Many scientific efforts have been performed to improve the streaming experience by optimizing the internals of HAS players. Such optimizations focus on the adaptation algorithms in HAS players to overcome bandwidth estimation errors, but they overlook the crucial role of the network in providing an environment where HAS players can operate without difficulty.

In this dissertation, we validate the hypothesis that the addition of a network-based control element that guides HTTP adaptive streaming clients improves the video quality and fairness between clients in shared networks. To this end, we develop, analyze, and optimize mechanisms and policies for bandwidth sharing between HAS clients. We combine empirical and theoretical research methodologies to gain a broad understanding of network optimizations for video streaming. Real implementations and experiments in different networks allow us to evaluate and improve the mechanisms for interaction between HAS clients and the network. We use Markov modeling to understand the impact of sharing policies on the
overall streaming performance, allowing faster results and more insights on the optimization of sharing policies that meet the requirements of the user.

**A network-based control element for HAS optimization**

No network elements providing the necessary flexibility and functionality exist that allow us to validate our hypothesis. In Chapter 2, we introduce the concept and two embodiments of the Control Element. The Control Element gathers information about the network and active HAS players, divides the network bandwidth among the HAS players according to a sharing policy, and translates bandwidth sharing decisions into traffic control rules. The first implementation of the Control Element is a HAS-aware proxy server, focusing on a transparent solution towards HAS clients. We describe the design of a proxy server that obtains HAS related information by monitoring HTTP requests. The proxy server controls HAS clients by modifying certain HTTP requests. The second implementation is a network element that maintains a control channel with HAS clients to obtain status information and provide bitrate recommendations. This implementation uses Software Defined Networking (SDN) techniques to control the network infrastructure. The Control Element improves over the related work, because it creates a network environment where HAS players stream at the optimal bitrate given the current network state.

To prove this concept, we describe a series of experiments that evaluate the Control Element. In Chapter 3, we compare the streaming performance of reference HAS implementations in best effort networks with HAS in networks optimized by our Control Element. We assess the two different implementations of the Control Element using wired and wireless testbeds, focusing on video freezes, video quality, changes in video quality, and fairness between players. Furthermore, we investigate the impact of different traffic control techniques on HAS, comparing different types of traffic control queues, varying the size of the queues, varying the number of queues in the network, and varying the number of HAS clients per queue. We show that our solution improves the streaming experience by preventing video freezes, minimizing the number of quality switches, and providing a fair video quality to all players. In experiments with up to 600 concurrent HAS clients, we show scalability and effectiveness of the Control Element in larger networks.

**Modeling network-assisted HAS**

While the evaluations in Chapter 3 prove the streaming performance increase when using the Control Element, implementation and experimentation is time consuming. In Chapter 4, we present a method for understanding and predicting the overall streaming performance given a bandwidth sharing policy in the Control Element. By formulating the starting and stopping of different types of players in a multi-dimensional Markov model that takes HAS characteristics into account, our model
predicts the video quality, the number of video quality switches, and fairness for each type of player. We perform a model-based analysis of sharing policies that include HAS players with different device characteristics and with premium users. In an extensive validation, comparing the model-based results with results from the experiments with the Control Element, we show the high accuracy of our model. This allows us to gain insights on the performance of HAS with the Control Element much faster.

More and more video is consumed via mobile networks. These networks have different characteristics in terms of throughput and resource allocation compared to fixed and Wi-Fi networks. This raises the question, how the Control Element can improve the video quality while also increasing the delivery efficiency. In Chapter 5, we present an extensive data set of LTE channel quality traces and formulate a Markov model for predicting the LTE channel quality in the near future. Based on this model, we formulate a strategy that combines resource allocation in the mobile network with buffering in the client, exploiting the fact that data transmissions under good channel quality are more efficient compared to transmission under poor channel quality. The strategy aims at proving a stable video quality, while minimizing the needed resources in the network. The parameters for the strategy are obtained by solving a Markov Decision Process. We show that we can stream video in a stable quality and increase the efficiency of the mobile network.