Implementing Application-Aware Resource Allocation on a Home Gateway for the Example of YouTube

Florian Wamser, Lukas Iffländer, Thomas Zinner, and Phuoc Tran-Gia

University of Würzburg, Am Hubland, Germany
{wamser,ifflaender,zinner,trangia}@informatik.uni-wuerzburg.de

Abstract. Today's Internet does not offer any quality level beyond best effort for the majority of applications used by a private customer. If multiple customers with heterogeneous applications share a bottleneck link to the Internet, this often leads to quality deterioration for the customers. Such a case occurs especially in home networks where broadband connections are still insufficient, or the rise of private wireless access networks leads to insufficient wireless resources. In this case, the best effort allocation of resources between heterogeneous applications leads to an unfair distribution of the application quality among the users. According to the principle of balanced application quality for all users, we propose to implement an application-oriented resource management on a home gateway. Therefore, allocation mechanisms need to be implemented such as the prioritization of network flows. Furthermore, a component monitoring the application quality and dynamically triggering these mechanisms is required. We show the feasibility of this concept by the implementation of an application monitor for YouTube on a standard home gateway. The gateway estimates the YouTube video buffers and prioritizes the video clip before the playback buffer depletes.

1 Introduction

The success of tablet computers, game consoles, and Smart TVs reflects the increased user demand for Internet-based services at home. The users in the home network can access value-added services offered directly by the network provider, such as IPTV. Likewise, they also use Over-The-Top (OTT) services like YouTube, Netflix, or online gaming and browse the web or download files. All these services have specific requirements with respect to the network resources which have to be fulfilled to ensure a good Quality-of-Experience (QoE) for the users. Furthermore, multiple users may concurrently access different services via the central Internet access point in the home network, the home gateway.

As stated by the Home Gateway Initiative (HGI) [1], the network at home and the broadband Internet access link may constitute a bottleneck. This may be due to the insufficient availability of broadband access, i.e., the network provider offers only smallband Internet access, or due to limitations within the home network, like the varying channel quality of the wireless networks.

Traffic in today's network structures is usually transmitted on a best effort basis. As a result, different services or applications with varying requirements and capabilities are treated equally on a per flow-basis, resulting in a unfairness in terms of QoE. Long lasting OS updates on a home computer may thus interfere with video streaming to a Smart TV which leads to video stallings and a degradation in the QoE of the user. In such a case, it is necessary to explicitly allocate the network resources unequally to the involved applications on network level to achieve a similar QoE across multiple applications.

This explicit network resource control is known as Application-Aware Networking. Scalability issues hinder the implementation of such mechanisms within aggregation and wide area networks, but the small size of the home network makes them a promising candidate. Most of the traffic is forwarded in home networks via the home gateway, making it possible to control the network resources, and therewith the application quality, at this entity.

In this paper, we show the potential of flexibly allocating network resources to different applications. We focus on a two-application scenario where YouTube flows and a file download compete for resources via a shared bottleneck link. YouTube maintains a playback buffer to overcome resource limitations on a short time scale. We take advantage of this in order to provide an accurate reaction against video stallings. We implemented a network-based buffer estimator for YouTube which allows the accurate monitoring of the application video pre-buffering state. If the local video buffer runs empty, the IP flow is prioritized, if the buffer is sufficiently filled, the prioritization is turned off, or other applications like browsing are prioritized.

The remainder of this paper is structured as follows. Section 2 summarizes the home gateway architecture and its components. In Section 3, the implementation details for the specific scenario are described. Section 4 highlights the evaluation setup, and Section 5 presents the results of our evaluation. The related work is summarized in 6, and the paper is concluded in Section 7.

2 Application-Aware Resource Control for a Home Gateway

In this section, we briefly discuss the application-aware resource management framework running on the home gateway and its components.

The developed architecture consists of four components. These are a network monitoring, a network control component, an application monitor, and a decision component. The purpose of the architecture is a cross layer approach that manages the network resources in a more sophisticated way in order to increase the QoE of all network users. It manages the network resources based on the application information. Resource management actions are only triggered if they may improve the average application performance of all network users. This means that they are triggered if an indication for, or the QoE degradation itself, has been detected and a resource management action may improve the QoE of

one application while not degrading the quality of others. In the following, we describe the four components.

Network Monitoring The network monitoring and control component firstly performs network monitoring of relevant QoS parameters and keeps track of the current network load. It provides a collection of relevant QoS parameters to the decision component. Thus, it supports the decision component by forcing appropriate resource management actions.

Network Control This component implements possible resource control mechanisms for an explicit allocation of network resources to single flows or traffic classes. With respect to the capabilities of the home gateway, these mechanisms may range from additional entries in the routing table to the prioritization of application classes or flows, sophisticated resource allocation actions or even an active manipulation of the application layer content. The network control component implements the input from the decision component.

Application Monitor The application monitor continuously estimates the current application quality based on the applications' network flows. The quality depends on the current application state which is a collection of application-specific quality indicators. These quality indicators describe whether the current resources offered by the network are sufficient for a good quality. The resulting application quality is forwarded to the decision component.

Decision Component The decision component sequentially determines how the available resource control mechanisms shall be forced based on the current network and application monitoring data. It predicts how the resource control may change the application and network state. A resource control action is triggered if the confidence in the prediction is high, the degree of the QoE improvement is significant, and the effort performing the resource management action is acceptable.

3 Implementation Details and Application Design

This section highlights the implementation details of our approach.

3.1 Monitoring the YouTube Buffer state

Details of our approach to estimate the application quality of YouTube are discussed in this subsection.

To enable a smooth playback of YouTube videos for the customer with a fluctuating network bandwidth, the YouTube application buffers a certain amount of the video data. To further achieve an economically efficient service, the prebuffered video data have to be kept as small as possible to minimize the waste of transport resources if the user aborts the video playback. This task is performed

by different flow control algorithms. Thus far YouTube implements two different approaches on flow control: the “throttling approach” performed on the server side by controlling network bandwidth [2] and the “range request approach” performed by the client by requesting the required parts of the video similar to the MPEG-DASH approach [3].

YouTube supports multiple media container formats [4]. Each format encapsulates the content provided in a different way. The most common formats are Flash Video (FLV) [5], Third Generation Partnership Project (3GPP) [6] and MPEG 4 (MP4) [7] where MP4 can be implemented as a continuous or fragmented format, the last one currently being the default format for newly uploaded videos to the platform.

MP4 specifies that media and meta data can be separated and stored in one or several files, as long as the meta data is stored as a whole. The file is divided in so-called boxes. The standard files contain one “ftyp”, “mdat” and “moov” box. Where “ftyp” specifies the file type, “mdat” contains the media data and “moov” the meta data. “moov” has serveral subboxes that contain the video header and timescale, information about substreams and the base media decode time.

Fragmented MP4 provides the ability to transfer the “mdat” box in multiple fragments instead of one big blop. As shown in Fig. 1 each fragment consists of a “moof” and a “mdat” box. The “moof” (movie fragment) box contains subboxes determining the default sample duration, the number of samples in the current track and a set of independent samples.

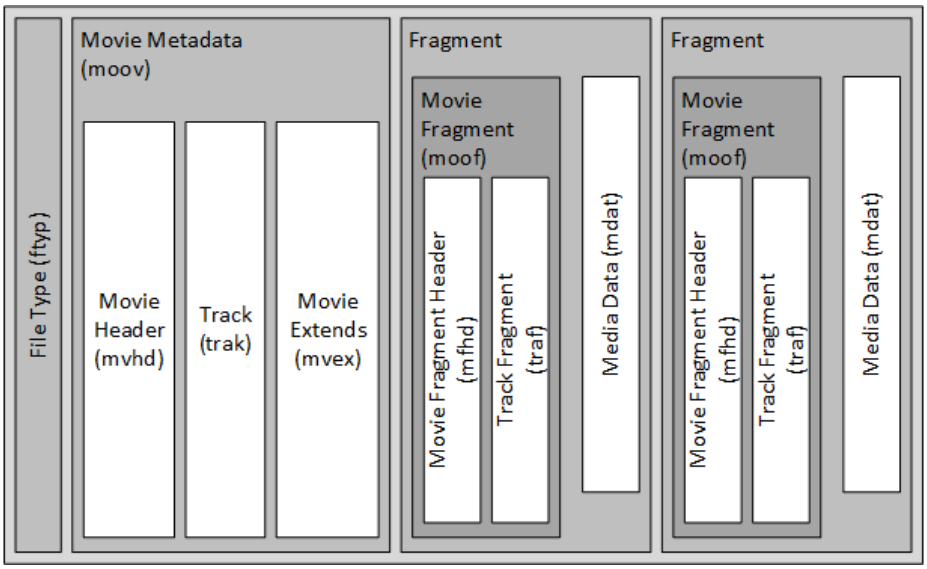

Fig. 1. MP4 fragment format

The fragmented MP4 is currently used for YouTube's range request algorithm and is currently the default format for newly uploaded videos. Therefore the application used for the YouTube buffer prediction focuses on this format.

To detect the requests to the server, the outgoing TCP traffic is monitored for HTTP GET requests. When a request is detected, video information like the video id, the format and the signature as well as streaming related information like the utilized flow control approach, are derived.

As for accumulating the currently downloaded amount of data for a certain video, the incoming traffic is monitored for the fragment boxes and incoming fragments are added to the estimation of the video's download process. The buffer estimation is then calculated as the difference between the available playback time of the already downloaded amount and the time passed since the begin of playback. We compute the application state for the video and the audio flow. User interaction like pausing, reverse jumping can not be taken into account. Forward jumping can be identified using the media data if the user jumps to a location which has not been downloaded yet.

3.2 Enforcing Resource Control

The most flexible way to control the network resources on a per-flow basis is the assignment of a dynamic rate limit to the individual flows. This enables a granular adjustment of each flow, but also results in a lot of state information required to maintain the single queues for each individual flow. A more simple approach is to define several priority queues and assign the flows to the different priorities with respect to the current application state, cf. Fig. 3. Since we evaluate a small home scenario, we implemented the priority approach by using five different queues with a queue size of 25 packets for each queue. To dynamically re-assign flows to different priorities, we use a Python wrapper script for the Linux Traffic Control [8] (TC) API via a simple TCP socket interface. With TC a stateful queuing is done that classifies the incoming packets and sorts them into the priority classes that are emptied by a scheduler according to the priorities.

3.3 Triggering Resource Control Actions

The network controller is implemented within the application monitor and executed periodically. If the YouTube video or audio playback buffer under runs 25 s, it is moved to a higher priority queue. If they overrun 40 s, the flow is moved to a lower priority. Thus, a reaction takes place if the application quality is endangered, i.e., the playback buffers threaten to run empty. Best-effort behavior is used, as long as the application quality for all applications is not endangered.

3.4 Application Design

The described components are implemented in two binaries. One is the resource enforcement application using priority queues. The other components are implemented as a C++ linux application. The multi-threaded design of the application

together with the compiler optimizations provided by C++ allow low latencies and a high performance on the used end device. We use the local TCP/IP interface to perform the communication between the applications. The redirection of the YouTube flow through the user space, however, has a significant impact on the data plane performance, since the CPU of the home gateway becomes the bottleneck. This results in a CPU load of 40% for a link capacity of 3 MBit.

4 Experimental Setup and Measurement Procedure

In the following subsection, the measurement scenario and the measurement sequence is described in order to evaluate the resource control.

4.1 Experimental Setup

The implemented application is designed to run on a typical home gateway. Of the available off the shelf gateway hardware, an AVM Fritz!Box 3390 has been chosen, featuring a MIPS 34Kc CPU running at 300 MHz with 22 MiB of RAM. To allow the usage of custom software and to provide the default linux libraries the alternative open source operating system Freetz [9] is used. On the Fritz!Box the resource control application and the combined decision and monitoring application are started.

The gateway is connected to the Internet via Ethernet with a theoretical bandwidth of 100 Mbit/s. In order to emulate a limited connection to the Internet, we restrict the downlink from the Internet to 3 Mbit/s. Two client laptops

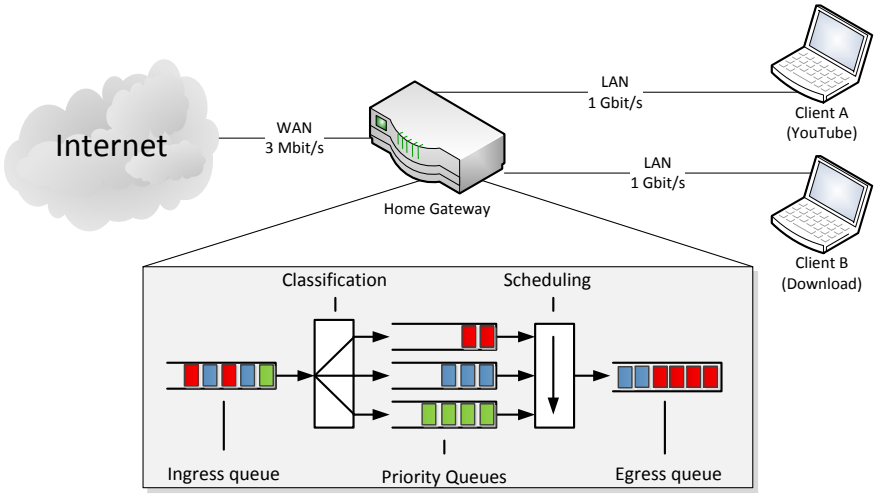

Fig. 2. Buffered Playtime for Audio and Video

(Client A and Client B) are connected to the gateway via Ethernet with a link speed of 1 Gbit/s.

4.2 Measurement Procedure

For our investigations, we use the following procedure. The scenario begins with starting a YouTube video clip on Client A. After 15 seconds a download is started on Client B utilizing four parallel connections for the download. Hence, we can expect the download to get at least four times the bandwidth of the YouTube stream. As video clip, we always use the same video clip,"VAN CANTO - Badaboom (Official)"[1] in 720p. After 300 seconds we stop the measurement.

5 Evaluation Results

5.1 YouTube Buffer Level During Video Playback

At first, we have a look at the YouTube buffer level over time without any resource management. Using fragmented MP4, the audio and the video stream are transmitted separately resulting in different playback buffer states. If one of the playback buffers is empty, the video clip stalls until the stalling threshold of 5 s is reached.

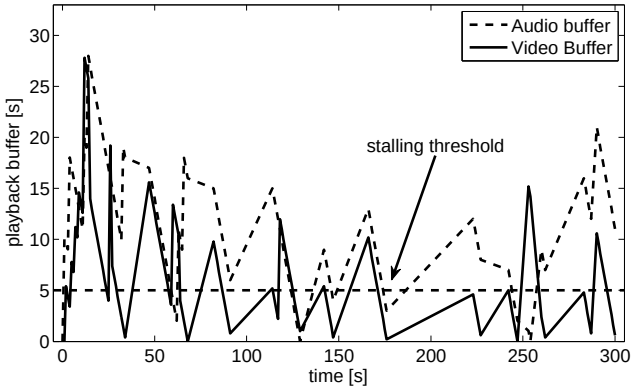

Fig. 3. Buffered Playtime for Audio and Video

Fig. 3 shows the playback buffer for the audio and video buffers. Both buffer levels increase until the parallel download is started. After that, the buffer level for both buffers decreases at about 20 s. In this example, the video buffer initially is empty at second 40, resulting in a short stalling period although the audio

[1] Video clip available at http://www.youtube.com/watch?v=Aeaz4s7q0Ag&wide=1&hd=1

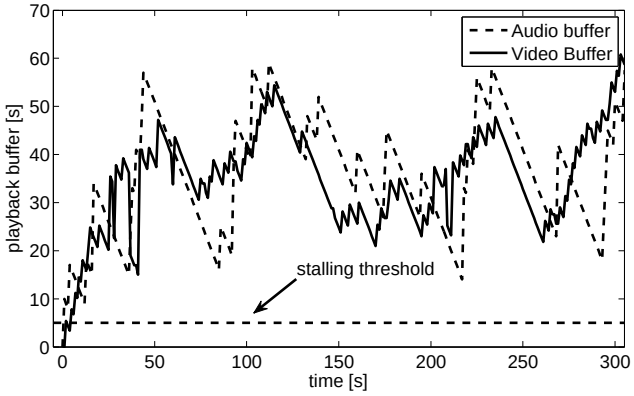

Fig. 4. Buffered Playtime for Audio and Video

buffer stays at a sufficient buffer level. Since not enough network capacity is available to allow a smooth video playback in case of a concurrent file download, the video playback is interrupted several times. After 60 s, the audio buffer underruns the threshold of 5 s while the video buffer still maintains a higher buffer level. Consequently, the audio flow is also taken into account in the following.

Fig. 4 shows audio and video playback buffer over time with dynamic prioritization enabled. In contrast to the best-effort case, the buffer levels increase although the parallel download has started. Both flows are prioritized until the playback buffers overrun a threshold of 45 s. After that, the resource control mechanism is turned off, until the buffers fall below 25 s. The TCP flow control results in additional dynamics and it may take some time until a fair bandwidth share between the flows is reached. This is reflected by the minimal and maximal playback buffer levels at 15 s and 63 s. In addition, the re-assignment to different priorities may lead to packet reordering influencing the TCP control loop, cf. [10].

5.2 Statistical Evaluation of the Investigated Scenario

After investigating the time series for the audio and video streams for the scenarios with and without prioritization, we focus on a statistical significant comparison between both approaches. For that, we conduct 10 runs and compare the CDFs for the buffer fillings. The results of this investigation is illustrated in Fig. 5 with a confidence level of 95 %. It can be seen that the application-aware approach using dynamic prioritization with respect to the application state clearly outperforms the approach without prioritization, i.e., the best effort case. This holds for both flows, the audio and the video flow. Stalling is minimized allowing a better user-perceived quality for the video streaming user. Further, it can be seen, that the buffer level for the audio buffer is typically higher as for the video buffer. Hence, we can conclude that a video stalling is more likely due to a video buffer under run. Fig. 5 also shows that the majority of values are

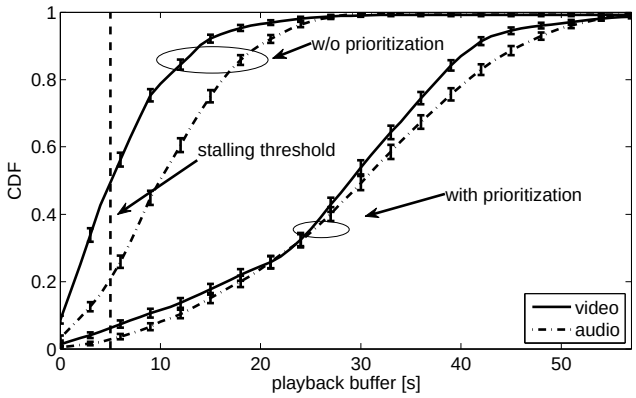

Fig. 5. Buffered Playtime for Audio and Video

located between 20 and 50 seconds, closely related to the values of the control hysteresis.

6 Related Work

In the following, papers on different resource management approaches are summarized. We specifically address QoE cross-layer resource management and application-aware resource management.

Different common techniques for resource management are used in networking devices and current communication protocols [11, 12, 13]. They all make use of protocol-specific conditions to favor packets or IP flows, or to influence the data transmission. To identify relevant packets or streams other techniques are necessary. Approaches like deep packet inspection (DPI) or explicit signaling of the application [14] are used to identify different applications and to map them to QoS classes. Approaches like [15, 16] tag the ToS (type of service), the diffserv (differentiated services) field in the IP packet or use a shim header [14] to prioritize the corresponding network traffic.

QoE Cross-Layer Resource Management The goal of QoE resource management approaches is a resource allocation according to the QoE of the users. QoE models [17, 18] for different applications form the basis as proposed in [19, 20, 21, 22, 23]. In order to incorporate QoE, a cross-layer optimization is typically used [19, 20, 23]. For example, in case of overload on the network layer, [19, 20] prioritize important packets of MPEG videos, mainly I-frames, in order to still guarantee a high quality. In [24], a multi-layer video encoding with scalable video codec is used. The goal is to prioritize the different layers in different QoS classes to specifically drop video layers with less importance if the network is congested. In [21], QoE-based scheduling for wireless mesh networks is proposed. According

to a MOS metric that maps QoE to QoS parameters, the video, data, and audio traffic is forwarded.

Application-aware resource management In addition to the work mentioned above, application-aware resource management is seen as a step towards QoE-oriented or QoE-aware resource management. It allocates resources based on dynamic cross-layer information from the application, e.g., the buffered playtime in case of videos in order to increase QoE [25, 26, 27]. This approach can be seen as a continuation of former work which used a so-called utility function for network resource management to define a scheduling order with respect to different application cases [28, 29, 30, 31, 32].

In [33], a QoS manager is proposed which centrally passes the requirements for the whole network to a switch. The problem is addressed how various networking entities can work together to prioritize traffic flows in order to allow a prioritization for the entire network. For residential networks, [34] proposes to setup a QoS system that uses hints from individual hosts to make decisions about traffic prioritization. In [35], the concept for traffic prioritization is presented for small networks. The authors specify a prioritization of applications for active users and a de-prioritization for applications in the background.

7 Conclusion

Traffic in today's network structures is typically transmitted on a best effort basis. As a result, different services or applications with varying requirements and capabilities are treated equally on a per flow-basis. This may result in unfairness in terms of the user-perceived quality, in particular if the overall resources are limited. This holds especially for home networks where different users may compete for limited network resources.

In this paper, we presented an application-aware networking approach to overcome this problem. We investigated the feasibility of the approach for a scenario consisting of two users, a download user and a user watching a YouTube video clip. We implemented an application monitoring component for YouTube, a prioritization mechanism to control the resources, and a simple decision logic. The monitoring component is able to estimate the video and audio playback buffer. The resulting information are used to trigger a prioritization of the video streaming if the buffer runs empty. Hence, a good QoE for the video streaming user can be guaranteed. The components were implemented on a typical home gateway. The evaluation of the scenario indicates the potential of the mechanism to manage the application quality and therewith the QoE for multiple users in a multi-application scenario.

References

1. Home Gateway Initiative, "Home Gateway QoS Module requirements," whitepaper, Dec. 2012.
2. Alcock, S. and Nelson, R., "Application flow control in youtube video streams," *ACM SIGCOMM Computer Communication Review*, vol. 41, no. 2, pp. 24–30, 2011.
3. C. Sieber, T. Hoßfeld, T. Zinner, P. Tran-Gia, and C. Timmerer, "Implementation and User-centric Comparison of a Novel Adaptation Logic for DASH with SVC," in *IFIP/IEEE International Workshop on Quality of Experience Centric Management (QCMan)*, Ghent, Belgium, May 2013.
4. Wikipedia, "Youtube — Wikipedia, the free encyclopedia," 2014, [Online; accessed 20-May-2014]. [Online]. Available: https://en.wikipedia.org/wiki/Youtube
5. A. S. Incorporated, *Video File Format Specification*, Adobe Systems Incorporated Std., Rev. 10, November 2008, retrieved 26 May 2014. [Online]. Available: http://download.macromedia.com/f4v/video_file_format_spec_v10_1.pdf
6. D. SINGER, *3GPP TS 26.244; Transparent end-to-end packet switched streaming service (PSS); 3GPP file format (3GP)*, ETSI 3GPP Std., Rev. 12.3.0, March 2014, retrieved 26 May 2014. [Online]. Available: http://www.3gpp.org/DynaReport/26244.htm
7. *MPEG 4 standards ISO/IEC 14496-1 ff*, International Organization for Standardization Std., Rev. 2010, 1999. [Online]. Available: http://www.iso.org/iso/iso_catalogue/catalogue_ics/catalogue_detail_ics.htm?csnumber=24462
8. B. Hubert, *Linux Advanced Routing & Traffic Control*, Linux Foundation, retrieved 26 May 2014.
9. Freetz project, "Freetz," http://freetz.org/.
10. T. Zinner, M. Jarschel, A. Blenk, F. Wamser, and W. Kellerer, "Dynamic Application-Aware Resource Management Using Software-Defined Networking: Implementation Prospects and Challenges," in *IFIP/IEEE International Workshop on Quality of Experience Centric Management (QCMan)*, Krakow, Poland, May 2014.
11. IEEE 802.1Q 2011, "Standard for Local and Metropolitan Area Networks - Media Access Control (MAC) Bridges and Virtual Bridge Local Area Networks," August 2011.
12. IEEE 802.11e-2005, "Standard for Information technology - Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) specifications," November 2005.
13. IEEE 802.16m-2011, "Standard for Local and metropolitan area networks - Part 16: Air Interface for Broadband Wireless Access Systems, Amendment 3: Advanced Air Interface (802.16m-2011)," May 2011.
14. S. Paul, R. Jain, J. Pan, J. Iyer, and D. Oran, "Openadn: A case for open application delivery networking," in *Computer Communications and Networks (ICCCN), 2013 22nd International Conference on*. IEEE, 2013, pp. 1–7.
15. "RFC 791: DARPA Internet program protocol specification," 1981.
16. K. Nichols, S. Blake, F. Baker, and D. Black, "RFC 2474: Definition of the differentiated services field (DS field) in the IPv4 and IPv6 headers," 1998.
17. F. Agboma and A. Liotta, "QoE-aware QoS management," in *6th International Conference on Advances in Mobile Computing and Multimedia*. ACM, 2008, pp. 111–116.

18. M. Fiedler, T. Hoßfeld, and P. Tran-Gia, "A Generic Quantitative Relationship between Quality of Experience and Quality of Service," *IEEE Network, Special Issue on Improving QoE for Network Services*, Jun. 2010.
19. J. Gross, J. Klaue, H. Karl, and A. Wolisz, "Cross-layer optimization of OFDM transmission systems for MPEG-4 video streaming," *Computer Communications*, vol. 27, no. 11, pp. 1044–1055, 2004.
20. S. Khan, Y. Peng, E. Steinbach, M. Sgroi, and W. Kellerer, "Application-driven cross-layer optimization for video streaming over wireless networks," *Communications Magazine, IEEE*, vol. 44, no. 1, pp. 122–130, 2006.
21. A. Reis, J. Chakareski, A. Kassler, and S. Sargento, "Quality of experience optimized scheduling in multi-service wireless mesh networks," in *IEEE Conference on Image Processing (ICIP)*. IEEE, 2010, pp. 3233–3236.
22. R. Pries, D. Hock, and D. Staehle, "QoE based Bandwidth Management Supporting Real Time Flows in IEEE 802.11 Mesh Networks," *Praxis der Informationsverarbeitung und Kommunikation*, vol. 32, no. 4, pp. 235–241, 2010.
23. P. Ameigeiras, J. J. Ramos-Munoz, J. Navarro-Ortiz, P. Mogensen, and J. M. Lopez-Soler, "QoE oriented cross-layer design of a resource allocation algorithm in beyond 3G systems," *Computer Communications*, vol. 33, no. 5, 2010.
24. C. Huang, H. Juan, M. Lin, and C. Chang, "Radio resource management of heterogeneous services in mobile WiMAX systems [Radio Resource Management and Protocol Engineering for IEEE 802.16]," *Wireless Communications, IEEE*, vol. 14, no. 1, pp. 20–26, 2007.
25. S. Thakolsri, S. Khan, E. Steinbach, and W. Kellerer, "QoE-Driven Cross-Layer Optimization for High Speed Downlink Packet Access," *Journal of Communications*, vol. 4, no. 9, pp. 669–680, Oct. 2009.
26. L. Superiori, M. Wrulich, P. Svoboda, M. Rupp, J. Fabini, W. Karner, and M. Steinbauer, "Content-aware scheduling for video streaming over HSDPA networks," in *Cross Layer Design, 2009. IWCLD'09. Second International Workshop on*. IEEE, 2009, pp. 1–5.
27. B. Staehle, F. Wamser, M. Hirth, D. Stezenbach, and D. Staehle, "AquareYoum: Application and Quality of Experience-Aware Resource Management for YouTube in Wireless Mesh Networks," *PIK - Praxis der Informationsverarbeitung und Kommunikation*, 2011.
28. M. Xiao, N. Shroff, and E. Chong, "A utility-based power-control scheme in wireless cellular systems," *Networking, IEEE/ACM Transactions on*, vol. 11, no. 2, pp. 210–221, 2003.
29. M. Andrews, L. Qian, and A. Stolyar, "Optimal utility based multi-user throughput allocation subject to throughput constraints," in *IEEE INFOCOM*, vol. 4. IEEE, 2005, pp. 2415–2424.
30. G. Song and Y. Li, "Utility-based resource allocation and scheduling in ofdm-based wireless broadband networks," *Communications Magazine, IEEE*, vol. 43, no. 12, pp. 127–134, 2005.
31. A. Saul, "Simple optimization algorithm for mos-based resource assignment," in *VTC Spring 2008. IEEE*. IEEE, 2008, pp. 1766–1770.
32. X. Pei, G. Zhu, Q. Wang, D. Qu, and J. Liu, "Economic model-based radio resource management with qos guarantees in the cdma uplink," *European Transactions on Telecommunications*, vol. 21, no. 2, pp. 178–186, 2010.
33. D. Zinca, V. Dobrota, C. Vancea, and G. Lazar, "Protocols for communication between qos agents: Cops and sdp." COST.

34. M. Katchabaw, H. Lutfiyya, and M. Bauer, "Usage based service differentiation for end-to-end quality of service management," *Computer communications*, vol. 28, no. 18, pp. 2146–2159, 2005.
35. J. Martin and N. Feamster, "User-driven dynamic traffic prioritization for home networks," in *Proceedings of the 2012 ACM SIGCOMM workshop on Measurements up the stack*. ACM, 2012, pp. 19–24.