

Improving Fairness for Adaptive HTTP Video Streaming

Bjørn J. Villa, Poul E. Heegaard¹ and Anders Insteffjord²

¹Department of Telematics,
Norwegian Institute of Science and Technology, N-7491 Trondheim, Norway
{bjorn.villa,poul.heegaard}@item.ntnu.no

²Vimond Media Solutions AS
N-5011 Bergen, Norway
anders@vimond.com

Abstract. This paper presents an analysis of a suggested method for improving fairness among competing adaptive HTTP video streams. The metrics used as fairness indicators are differences in achieved average rate and stability among the competing streams. The method analyzed is based on changing a fixed and equal video segment request rate of each stream, to either a per session unique or random request rate. The analysis is done by means of measurements in a controlled environment using the Microsoft Smooth Streaming solution. The findings are considered very positive as they show that it is possible to achieve a significant improvement in fairness by applying the suggested method. The main novelty of the paper is that it demonstrates the potential of achieving such improvements without modifying either client or server algorithms.

Keywords: Fairness, Adaptive Streaming, Available Bandwidth Estimation

1 Introduction

Internet services with video components have become a great success in terms of general usage and also as facilitator of new business models on the Internet. A significant part of these services is carried as best-effort traffic between a content provider and the end user. In order to make this possible in a way which is acceptable to the user, a new concept of adaptive video streaming has emerged. The concept is used in solutions from many vendors (Microsoft, Apple, Adobe et.al.) – and they all share the same characteristic in terms of being able to adapt the video quality level during the session according to certain observed metrics (e.g. available bandwidth and CPU load). The main rationale behind this is to enable a video session to survive the varying conditions experienced on the Internet.

Even though the adaptive streaming solutions in the marketplace today share some important characteristics, they remain highly proprietary awaiting the successful completion and adoption of the ongoing active standardization effort known as MPEG DASH [1]. The scope of the standard covers interfaces and message formats, which will enable future interoperability between client and servers from different vendors.

Although the new standard will resolve the important interoperability issue, it will not remove the competitive differences between the solutions offered which reflects their approach and set of priorities. As an example, one could look at the solutions from Apple and Adobe which to some extent represent two opposites [2]. The Apple solution tries to maintain a more stable quality (minimize number of changes), at the expense of average quality level. The Adobe solution does it quite differently and almost continuously adjusts the quality level, giving a higher average quality level but at the expense of a potentially higher number of quality changes. The Microsoft solution [3] is somewhere between these two and tries to achieve stability and also a high average quality level, and this is the one used as reference point for the work reported in this paper.

1.1 Problem Statement

As the concept of adaptive streaming is becoming widely adopted for delivery of video services on the Internet there are weaknesses appearing. In fact, one could say that the success of the concept becomes its biggest problem. Reason being that the concept works best in a scenario when you are primarily concerned with one video service, and the rest of the traffic is of less importance. However, when multiple adaptive streaming sessions appear at the same time – competing for the same access capacity their performance becomes quite questionable. Traditional topics like intra flow fairness become an issue as well as session stability.

The most likely reason for performance degradation under such conditions is that the algorithms controlling the streams were not made with this in mind. An important area where improvements could be made is with regard to how each session estimates available bandwidth. Ideally, each session should in some way be able to get a notion of all other traffic and thereby also an accurate view on available bandwidth. A candidate reason for why this is not working properly in current solution may be the strict periodic burst/idle nature of each adaptive streaming session (cf. Fig. 1).

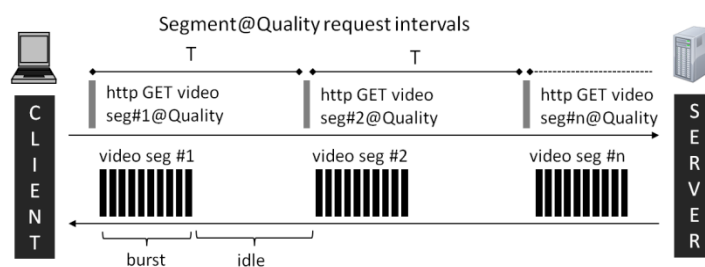


Fig. 1. Burst and idle periods of adaptive http streaming

The periodic characteristic is given by the per session segment@quality request intervals while the burst part is given by the fact that the server always tries to deliver the next segment as fast as possible in order to maintain a high degree of client buffer filling. For the Microsoft Smooth Streaming solution, the typical segment@quality

request interval T has a value of 2 seconds. Depending on which quality level a client session is running at it will then be receiving IP packets belonging to the next video segment only during parts of this period T . We call this part the burst period, and the remaining part of T the idle period.

A potential reason for errors made when calculating available bandwidth during each burst period is the fixed and identical periodic behavior of each session. In a situation where the competing traffic is of similar type, e.g. multiple adaptive video streams – we will have that each stream during its burst periods always will notice some of the other streams but at the same time always miss some others. This makes it challenging for the available bandwidth algorithm to provide quality input to the decision making process in the client. The result will be that in many cases the client will make a decision to go up in quality level, even though there is not enough available capacity. This again leads to an increased congestion with a potential overall session impact.

1.2 Research Approach

A suggested method of improving fairness among competing adaptive video streams was introduced in our previous work [4] and analyzed by means of simulations. This method was based on changing the fixed T parameter to a random T parameter according to some stochastic distribution.

In addition to this, we now also include new version of the method being the case where each competing stream is set to have a fixed but unique T value. The uniqueness applies among the group of streams competing for a specific amount of bandwidth, e.g. simultaneous streams delivered to members of a household. The purpose of both these suggested changes for the T value is to improve the accuracy of the available bandwidth estimation done by each session.

The approach taken in the work reported in this paper is to take the method validation one step further, by means of performing an experimental validation through a series of measurements in a controlled environment.

1.3 Paper Outline

The structure of this paper is as follows. Section 2 provides an overview of related work; Section 3 presents the measurement setup; Section 4 provides the results and an analysis; Section 5 provides the conclusions and an outline of future work is given in Section 6. Acknowledgements are given in Section 7.

2 Related Work

The amount of research in this field has been growing over the last years in parallel with commercial deployments and an ongoing standardization effort [6]. The most related research is found in the area of streaming performance evaluation and how to enhance key functionality related to session quality control.

In [7] the authors describe and evaluate selected commercial and open source players for adaptive streaming. They also perform an experimental evaluation of their performance. One of their interesting findings is that competing adaptive streams can cause unpredictable performance for each stream, both in terms of oscillations and ability to achieve fairness in terms of bandwidth sharing. In [8] the authors have investigated how well adaptive streaming performs when being subject to variable available bandwidth in general. Their findings were that the adaptive streams are performing quite well except for during some transient periods. These findings do not contradict the findings in [7] as the type of background traffic used did not have the adaptive behavior itself. In [2] the authors have performed an experimental evaluation of commercial adaptive streaming players in a challenging emulated 3G network environment. Their findings were that there are significant differences between the players in terms of behavior, which supports the findings in [7] but under different network conditions.

In [9] a rate adaptation algorithm which uses a smoothed HTTP throughput measure based on the segment fetch time is proposed and analyzed by means of simulation. This work was taken further by the same authors in [10] and then focusing on the importance of the video segment length and potential of choosing more optimal values for this. Further on, the same research group investigated the potential in using a parallel HTTP streaming approach [11] in order to provide a higher quality. Their simulation results demonstrated a significant potential in doing so. The authors of [12] have also suggested an enhanced method for performing available bandwidth estimation and verified it through experiments. The principle applied is the same as in [9] in the sense that measurements are done based on the data received for each video segment in the stream.

While many publications focus on how to make the client side of an adaptive streaming session more intelligent, the authors of [13] has taken to opposite approach and suggested to make the server side of it more intelligent. Their approach is intended to maintain a low complexity in the player, while at the same time achieving a high degree of effectiveness in session control. Alternative approaches are described in [14] and [15] where the role of the home gateway is investigated in terms of ability to improve adaptive streaming performance.

3 Measurement Setup

In order to perform the required measurements a testbed was established in a controlled environment including all required components. As illustrated in Fig. 2 a number of clients behind an access congestion point were set up to access a Microsoft Smooth Streaming service. The clients were accessing the same adaptive stream from the server and run in a loop with intervals of 25 minute active streaming and then 5 minute break. For each scenario studied the loop was set to give 100 interval repetitions. An earlier developed tool for event reporting [5] from each client (Monitor Plane event reports) is used in order to record interesting events on a per

session basis and allow for post processing. The available quality levels for the video stream used were 350, 500, 1000, 1500, 2000, 3000, 4000 and 5000Kbps.

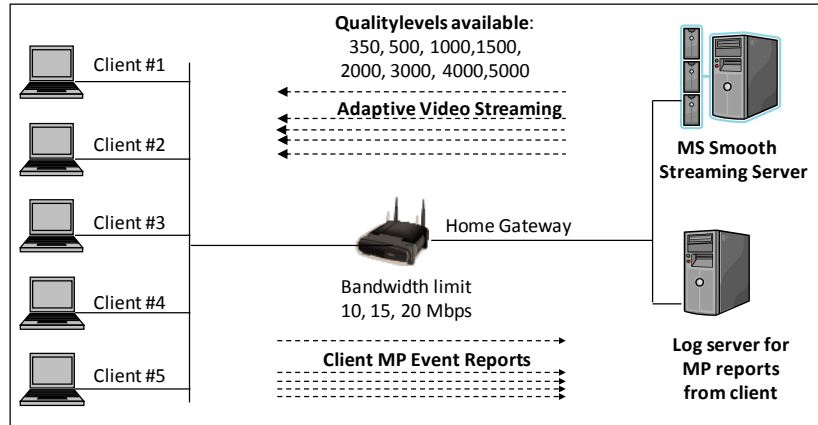


Fig. 2. Measurement testbed

A series of measurement scenarios were defined in order to study the effect of the suggested method under various conditions. The parameters describing a specific scenario was the access capacity shared between the competing adaptive streaming sessions and a specific segment@quality request interval approach (i.e. the T parameter). For the access capacity, values of 10, 15 and 20Mbps were used. This would all be cases with a high degree of congestion as the maximum per session video quality rate was 5Mbps. Thus, all sessions running at max quality rate would not be possible.

With regard to the T parameter, values between 1.6 and 2.5 sec with increments of 0.1 sec were available for use. This differs to some extent from the scenario studied in previous simulations [4] where increments of 0.01sec were available, but was the closest approximation the equipment used allowed for. The following cases were then considered, whereas the first is the reference point (Microsoft Smooth Streaming).

Table 1. Video segment request intervals used in experiments

Fixed T	T=2 sec	default value
Fixed T	T=1.6 sec	Lower than default
Fixed T	T=2.4 sec	Higher than default
Unique T	T ₁ =1.6, T ₂ =1.8 T ₃ =2.0 T ₄ =2.2 T ₅ =2.4	Each session different
Random T	T _i =Random{ 1.6-2.5}, steps of 0.1	Random select

In order to support the objective of validating the effect on fairness among the competing streams, both per session average rate and average number of rate reductions per minute were collected.

4 Results

The results are presented using a graphical view of the sample five-number summaries: the smallest observation, lower quartile, median (and mid 50% samples), upper quartile, and largest observation. The samples come from the measurement intervals as defined in section 3.

The samples for each interval are sorted and then grouped across all intervals according to their order. Statistics are then computed for a specific metric based on the grouped samples. This gives a view on the studied metrics to appear in a sorted manner from the worst to the best performing session, which implicitly also presents the differences between the sessions in a good way.

4.1 Average Session Rate

The distribution of average rate samples for the 5 competing sessions in the case of changing the fixed T value from its default value, to a higher or lower is presented in Fig. 3. The shared access capacity is 10Mbps.

The results show that the effect of changing the fixed T value to a lower or higher value does not have significant effect on the average session rate distributions. The sessions appear to spend most of their time at quality levels 1500Kbps/2000Kbps, and the best/worst performers are approximately within the same bounds as for the default case (T=2.0).

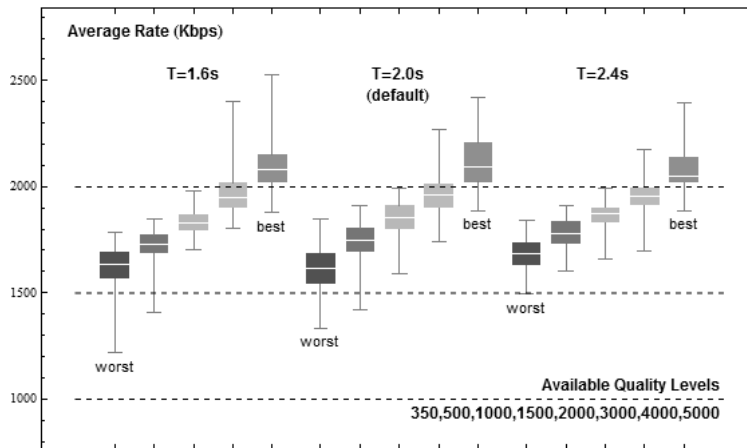


Fig. 3. Average Rate at 10Mbps and T=1.6/2.0/2.4

The distribution of average rate samples for the competing sessions in the case of changing the fixed (and equal) T value from its default value, to either a unique T value or a random T value per session is presented in Fig. 4.

The results show that a unique or random T value give distributions for the best performers with lower medians and for the worst performers it gives higher medians.

The spread of the median part of the distributions are also smaller. Considering also the outliers in the distributions, it appears that the unique T is better than the random T approach.

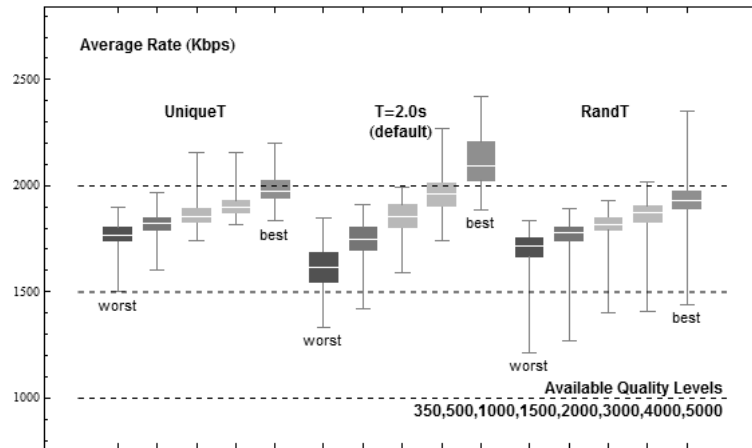


Fig. 4. Average Rate at 10Mbps and T= Unique/2.0/Random

4.2 Session Quality Reductions

The distribution of quality reduction samples for the competing sessions in the case of changing the fixed T value from its default value, to a higher or lower value is presented in Fig. 5.

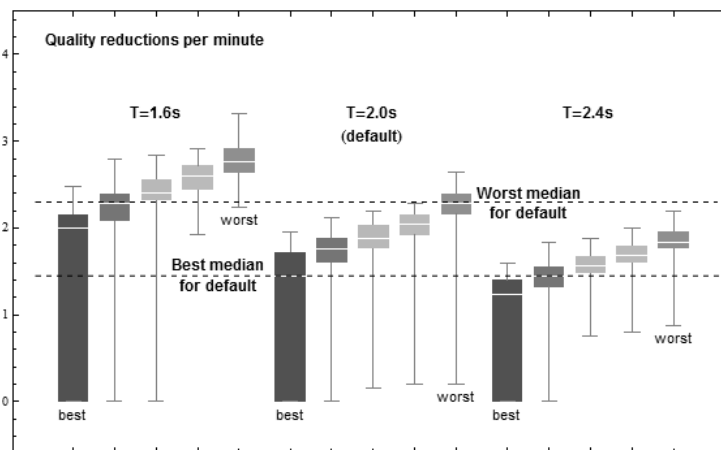


Fig. 5. Quality Reductions at 10Mbps and T=1.6/ 2.0/ 2.4

The results in Fig. 5 show that changing the fixed T value to a lower or higher value has a logical effect on the frequency of quality reductions. With a lower T

value, we get more reductions and with a higher T value we get less. This makes sense as the lower T value sessions have the opportunity to change quality level more frequently than higher T value sessions.

The distribution of quality reduction samples for the competing sessions in the case of changing the fixed (and equal) T value from its default value, to either a unique T value or a random T value per session is presented in Fig. 6.

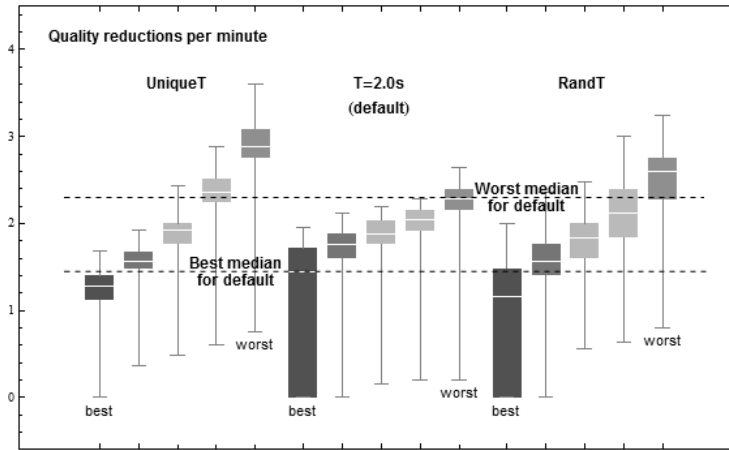


Fig. 6. Quality Reductions at 10Mbps and T= Unique/2.0/Random

The results in Fig. 6 show that the method of either using a unique or random T value give distributions for the best performing sessions with lower medians, and higher medians for the worst performing ones. The difference in medians for the best and worst performing sessions is higher for both the unique T and random T cases. However, the absolute values are small. In all cases, the number of quality reductions per minute is mainly between 1 and 3. Thus, the change is not considered significantly negative.

4.3 Session Fairness

The two metrics studied in the measurements are both relevant measures for intra-session fairness. The average rate metric is the traditional approach, where the optimal case is considered as the one where each session gets an equal share of the available bandwidth. The quality reduction metric is not so commonly used as a measure of fairness, but we have presented this as a candidate component of perceived fairness in our earlier work [4]. Based on this, we use both these metrics as basis for our evaluation of whether our suggested method of improving fairness for competing adaptive HTTP video streams is effective.

The measurement presented in section 4.1 and 4.2 were for the specific access capacity of 10Mbps as the shared resource for which the sessions as competing. In the following we present combined results including the access capacity levels of 15Mbps and 20Mbps.

In Fig. 7 an average rate fairness view is given by the spread of medians for each method (Unique T, Fixed T, Random T) at the three different access capacity levels. It is desirable to have a small difference between the median of the best performer and the worst performer, and also that they are close to the available quality level representing the optimal fair share. The capacity levels used in the measurements were chosen so that there would always be a valid quality level for the fair share as indicated in the figure.

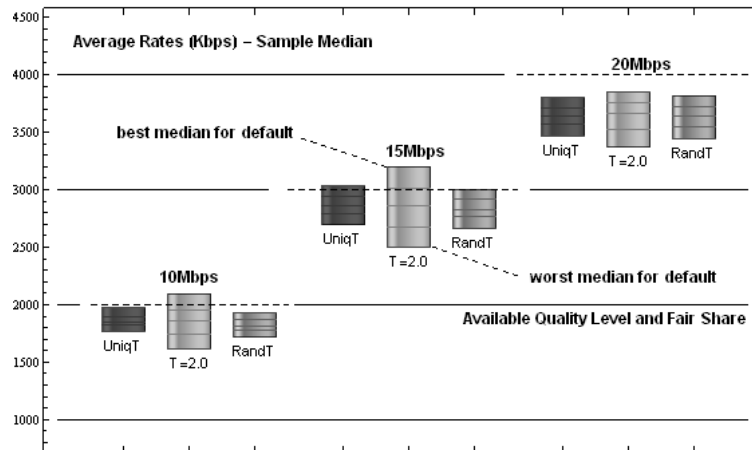


Fig. 7. Average Rate Fairness

The results in Fig. 7 show that for both the unique T and random T methods a significant improvement in intra-flow fairness is achieved. The improvement is highest at the two lowest capacity levels where the best/worst performer median difference is reduced by more than 50% while remaining close to the optimal fair share. For the highest capacity level the improvement is lower but still between 20-30%.

Table 2. Quality Reductions view on Fairness

Median	10Mbps Max-Min	15Mbps Max-Min	20Mbps Max-Min
UniqueT	1,60	1,28	1,20
FixedT	0,84	1,08	0,64
RandomT	1,44	1,08	1,04

Looking at fairness among the sessions from a quality reduction per minute point of view, the Max-Min median difference for the unique T, fixed T and random T methods are very close to each other. Therefore, it does not seem as if the significant improvement in average rate fairness come at the expence of stability in general or increased differences among the sessions.

5 Conclusions

The results from the measurements clearly show that the suggested method has a positive effect on fairness among competing adaptive video streams in terms of reducing the difference in average rate per session among competing sessions, without increasing the amount quality reductions significantly. For the two alternative approaches for setting the T value (unique or random) the results are very close to each other, but the unique T approach are slightly better and at the same time a less demanding approach than the random T approach.

A potentially relevant aspect not included as a variable parameter in the measurements is the significance of granularity in video quality levels. Intuitively, having a fine granularity would be ideal – but again this would increase production cost in terms of time and storage requirements.

The value of the findings should be considered most relevant for adaptive HTTP video streaming solutions which embed similar characteristics to the one used in the measurements from Microsoft, but also other adaptive services with a periodic and burst oriented nature. In terms of applying the findings to a real life solution, this is possible to do in different ways. One could of course make all video streams available with a range of different segment sizes and then communicate through the initial session manifest file [3] which one a specific client should use. However, this would be rather demanding from a content storage perspective. Alternatively, one could apply different segment sizes for different video streams and then get much of the same effect assuming that competing streams are not the exact same video stream. This is considered as a likely scenario for users in a home network, where users typically are somewhat different in terms of preferred content. This approach also has a benefit in terms of that it does not require duplication of the content and thereby no increase in storage requirements.

6 Future Work

For further enhancements of the performance for competing adaptive HTTP video streams it is likely that one should consider making available bandwidth estimations not only based on the data sent during each burst period, but also have some probing done during the idle periods. Looking closer into the potential effect of introducing some degree of memory in the algorithms controlling the adaptive streams may also provide interesting results. These issues are part of our planned future work, as well as making an integrated analysis of the suggested method using updated results from the earlier developed simulation model [4] together with measurement results.

7 Acknowledgements

The reported work is done as part of the Road to media-aware user-Dependant self-adaptive NETWORKS - R2D2 project. This project is funded by The Research Council of Norway.

References

- [1] ISO: Dynamic adaptive streaming over http (dash). iso/iec fcd 23001-6 (2011)
- [2] Riiser, H., Bergsaker, H.S., Vigmostad, P., Halvorsen, P., Griwodz, C.: A comparison of quality scheduling in commercial adaptive http streaming solutions on 3G network. In: Proceedings of the 4th Workshop on Mobile Video. MoVid '12, New York, NY, USA, ACM (2012)
- [3] Zambelli, A.: IIS smooth streaming technical overview. Technical report (March 2009, last accessed 10.01.2012)
- [4] Villa, B.J., Heegaard, P.E.: Improving perceived fairness and QoE for adaptive video streams. ICNS 2012 (March 2012)
- [5] Villa, B.J., Heegaard, P.E.: A monitor plane component for adaptive video streaming. In: NIK2011, Norway. ISSN 1892-0713. Volume 1. (Nov 2011)
- [6] Sodagar, I.: The mpeg-dash standard for multimedia streaming over the internet. *Multimedia*, IEEE 18(4) (april 2011)
- [7] Akhshabi, S., Begen, A.C., Dovrolis, C.: An experimental evaluation of rate-adaptation algorithms in adaptive streaming over http. In: ACM Multimedia Systems (MMSys). (2011)
- [8] De Cicco, L., Mascolo, S.: An experimental investigation of the Akamai adaptive video streaming. In: Proceedings of the 6th international conference on HCI in work and learning, life and leisure: workgroup human-computer interaction and usability engineering. USAB'10, Berlin, Heidelberg, Springer-Verlag (2010)
- [9] Liu, C., Bouazizi, I., Gabbouj, M.: Rate adaptation for adaptive http streaming. In: the second annual ACM conference Proceedings of the second annual ACM conference on Multimedia systems - MMSys 11, San Jose, CA, USA New York, New York, USA, ACM Press, ACM Press (2011)
- [10] Liu, C., Bouazizi, I., Gabbouj, M.: Segment duration for rate adaptation of adaptive http streaming. In: Multimedia and Expo (ICME), 2011 IEEE International Conference on. (July 2011)
- [11] Liu, C., Bouazizi, I., Gabbouj, M.: Parallel adaptive http media streaming. In: Computer Communications and Networks (ICCCN), 2011 Proceedings of 20th International Conference on. (31 2011-aug. 4 2011)
- [12] Mok, R.K.P., Luo, X., Chan, E.W.W., Chang, R.K.C.: Qdash: a QoE-aware dash system. In: MMSys. (2012)
- [13] De Cicco, L., Mascolo, S., Palmisano, V.: Feedback control for adaptive live video streaming. In: Proceedings of the second annual ACM conference on Multimedia systems. MMSys '11, New York, NY, USA, ACM (2011) 145–156
- [14] Houdaille, R., Gouache, S.: Shaping http adaptive streams for a better user experience. In: MMSys. (2012)
- [15] Villa, B.J., Heegaard, P.E.: Towards knowledge-driven QoE optimization in home gateways. ICNS 2011 (May 2011)