Improving Fairness in QoS and QoE domains for Adaptive Video Streaming

Bjørn J. Villa
Department of Telematics
Norwegian Institute of Science and Technology
Trondheim, Norway
bjorn.villa@item.ntnu.no

Poul E. Heegaard
Department of Telematics
Norwegian Institute of Science and Technology
Trondheim, Norway
poul.heegaard@item.ntnu.no

Abstract - **This paper presents an enhancement to a category of Adaptive Video Streaming solutions aimed at improving both Quality of Service (QoS) and Quality of Experience (QoE). The specific solution used as baseline for the work is the Smooth Streaming framework from Microsoft. The presented enhancement relates to the rate adaptation scheme used, and suggests applying a stochastic or fixed/unique setting of the rate adjustment intervals rather than the default fixed/equal approach. The main novelty of the paper is the simultaneous study of both network oriented fairness in the QoS domain and perception based fairness from the QoE domain, when introducing the suggested mechanism. The method used for this study is by means of simulations, measurements and numerical optimization. Perception based fairness is suggested as an objective QoE metric which, requires no reference to original content. The results show that the suggested enhancement has potential of improving fairness in the QoS domain, while maintaining perception based fairness in the QoE domain.**

Keywords - Adaptive Video Streaming; Fairness; QoE/QoS.

I. Introduction

Solutions for Adaptive Video Streaming are part of the more general concept of ABR (Adaptive Bit Rate) streaming [1] which, covers any content type. The implementation of ABR streaming for video varies between different vendors, and among the more successful one today is the Microsoft Smooth Streaming framework [2]. In general, the different implementations use undisclosed and proprietary functions, even across interfaces between client and server. The latter is addressed by the new MPEG DASH standard [3].

The basic behavior of adaptive video streaming solutions is that the client continuously performs a measurement and estimation of available resources in order to decide which, quality level to request. The relevant resource from the network side is the available capacity along the path between the server and client. Based on this, at certain intervals the client decides to either go up or down in quality level or remain at the current level. The levels are predefined and communicated to the client by the server at session startup. The changes in quality levels are normally done in an incremental approach, rather than by larger jumps in rate level. The rationale behind this is the objective to provide a smooth watching experience for the user. However, it may also be related to the CPU monitoring done by the client, as this is a key resource required. It may be the case that even if the network can provide you with a much higher rate level, the CPU on the device being used would not be able to process it. During the initial phase of an adaptive streaming session the potential requests of change in rate level are more frequent than later on when operating in a more steady-state phase. To some extent this is a rather aggressive behavior from a single client which, may have undesirable inter-stream impacts. At the same time, in order to give the user a good first impression and make him want to continue using the service it is desirable to reach a high quality as soon as possible.

Among the strongest drivers for commercial use of ABR based services on the Internet are Over-The-Top content providers. These are providers which, rely on the best effort Internet service as transport towards their customers. Therefore, technologies aiming at making services survive almost any network state are of great interest. In addition to focus on the network based QoS dimensions of services and involved networks, there is also a growing interest in the QoE dimension [4]. The latter should be considered as not only a richer definition of quality, but also more focused towards who decides whether something is good or bad, i.e., the end user. The evolution of successful services on Internet indicates that the focus on QoE for Over-The-Top providers is a good strategy.

A. Problem Statement

The concept of Adaptive Video Streaming is very promising. However, as more and more services are adopting this concept the success brings new challenges. The first challenge with effects visible to the end users is how well these services behave when they compete for a shared resource, such as a home broadband access. With a strong dominance of video based service on the Internet this issue is important to address. As each client operates independently of each other, it has no understanding of the traffic it competes with. Different clients consider each other as just background traffic. This leads to unpredictable behavior of each session. The focus of this paper is to study a method for improving QoS/QoE fairness among competing streams in a home network environment.

B. Research Approach

The method investigated in this paper to address the problem at hand is to apply specific changes in the algorithm used by each ABR client controlling the adaptive behavior. The specific change suggested is related to the rate adjustment interval used [2]. The effect of changing the duration of the rate adjustment interval from an equal T duration to either a random or per session unique duration is presented and analyzed.

The ABR solution used as reference point for the work is the one from Microsoft (Smooth Streaming). However, the key principles would still apply to other solutions based on similar principles.

C. Paper Outline

The structure of this paper is as follows. Section II presents related work; Section III provides an overview of methodology and metrics; Section IV describes the simulation model; Section V presents simulation results; Section VI presents the measurement setup together with results; In Section VII the simulation results and measurements results are summarized and compared; In Section VII an analytical view of the methods studied are given; Section IX provides the conclusions and an outline of future work.

II. Related Work

It has been shown in [5] 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. The experimental results presented give clear indication on that competing ABR clients cause degraded and unpredictable performance. Apart from this paper, the topic at hand does not seem to have been addressed by the academic research community to the extent it deserves.

In another paper [6], 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 in this type of scenario except for some transient behavior. These findings do not contradict the findings in [5] as the type of background traffic used do not have the adaptive behavior itself, but is rather controlled by the basic TCP mechanisms.

Rate-control algorithms for TCP streaming in general and selected bandwidth estimation algorithms are described in [7]. This work is relevant to any TCP based application delivering a video stream.

In some of our own previous work we have described and analyzed how competing adaptive streams can be controlled using a knowledge based bandwidth broker in the home gateway [8] [9]. We have also developed a testbed for performing experimental verification of methods studied [10] which has been used for collecting the measurement used in this paper.

III. Methodology and metrics

In this section, we introduce the relevant performance metrics together with motivation for the chosen focus. Thereafter, some candidate methods on how to improve the performance metrics are given, and finally, the specific method subject for study is presented.

A. Flow Based Performance Metrics

For transport flows it is common [11] to focus on the following metrics in order to assess their performance: inter-flow fairness, stability and convergence time. This in addition to the general QoS metrics: bandwidth, packet loss, delay and jitter. The same metrics can be applied to adaptive video streams as they by definition also are flows with similar concerns. The analysis of these metrics can be done from a strict network oriented perspective (QoS), but to some extent also bridged over to a user perception domain (QoE). When focusing on the inter-flow fairness metric this is traditionally analyzed [12] using, e.g., the Jain's fairness index [13], the product measure [14] or Epsilon-fairness [15] for flows with equal resource requirements. For flows with different resource requirement, the Max-Min fairness [16], proportional fairness [17] or minimum potential delay fairness [18] approaches are commonly seen. Real life adaptive video streams would typically belong to the last category.

Max-Min fairness: The objective of max-min fairness is to maximize the smallest throughput rate among the flows. When this is met, the next-smallest throughput rate must be as large as possible, and so on. Max-min fairness can also be explained by considering it as a progressive filling algorithm, where all flows start at zero and grow at the same pace until the link is full. With this approach the max-min fairness gives priority to the smallest flows. The least demanding flows always have the best chance of getting access to all the resources it needs.

Proportional fairness: The original definition of proportional fairness comes from economic disciplines [17] for the purpose of charging. The original definition is used in the relevant RFC [12] but it does not come across as very constructive for the purpose of analyzing fairness in single resource (e.g., bandwidth) sharing among flows. In this context more recent definitions and interpretations are more suitable [19]. The principle of this would be that a resource allocation is considered proportional fair if it is made to the flow which, has the highest ratio between potential maximum resource consumption and its average resource consumption so far. A further simplification would be to use the current resource usage (if greater than 0) instead of the average in the ratio calculation. The same ratio numbers for each flow could then be used to give a view on the current system fairness by comparing them. If they are all equal the system could be stated as proportionally fair.

Minimum potential delay fairness: The idea behind minimum potential delay fairness is based on the assumption that the involved flows are generated by applications transferring files of certain sizes. A relevant bandwidth sharing objective would be to minimize the time needed to complete those transfers. However, this does not apply to an adaptive streaming scenario and is therefore not discussed any further.

B. *Perception Based Performance Metrics*

There is a wide range of metrics which, influence how satisfied an end user is with a service such as e.g., video streaming. Many of these are not related to network aspects, and therefore difficult to influence by means in this domain. However, one of the perceived performance metrics which, could be correlated with network aspect is the notion of perceived fairness. It is then of great interest to try and find methods of influencing this in a positive manner.

Looking at fairness from an end user perception, research from the social science and psychology domain [20] states that this is closely related to what is called 'Social Justice'. In this context a queuing system or any other resource allocation mechanism would be considered as a 'Social System'. It has further been found that users react negatively to any system behavior which, gives better service to other users, unless justification is provided. Such system behavior is considered un-fair, i.e., in violation with the social justice of the system as the end users considers it as discrimination.

The end user notion of system discrimination has been suggested by [21] as an important measure of perceived service quality, and more specifically the perceived fairness is stated to be closely related to the discrimination frequency. It should be noted that analyzing this type of end user perceived discriminations has a challenge in terms of handling the false positive and false negative cases.

Applying the concept of discrimination to competing adaptive streams, it would be related to situations where end user expectations are not met during steady state periods and also negative changes in service delivery during more transient periods. In other words, whatever makes the end user think that he is being discriminated due to other users in the system, will lead to reduced perceived service quality.

In order to use this type of perceived end user discrimination as a measure for how well the algorithm which, controls the adaptive streams are performing, a clear definition regarding what end users are considering as discrimination is required. This could, e.g., be periods with session rate below some threshold, any change in session rate to a lower level or the session rate change frequency.

C. *QoS and QoE Fairness*

Based on the overview given in the previous sections for both flow based and perception based performance metrics, the following definitions are presented for the fairness metrics subject for study in this paper.

In the QoS domain, we use proportional fairness as the key metric while in the QoE domain we use perceived fairness, defined as follows.

Proportional Fairness - The difference between the worst and best performing streaming sessions in terms of average rate achieved during the session lifetime divided by session max rate.

Perceived Fairness – The difference between the worst and best performing streaming sessions in terms of average number of rate reductions (i.e. discrimination events) per minute.

Following this, the main focus is put on differences in performance for the worst and best performing sessions. However, the absolute values for both achieved session rate and session quality level reductions are of course also relevant when evaluating the proposed methods.

D. *Methods for Improving Performance*

There are several things that one could try to incorporate into the adaptive algorithms controlling the ABR service in order to make them perform better in a multi-stream scenario.

The selected performance metrics to be studied are proportional fairness and perceived fairness metric as described. Whether it is possible to improve both these fairness metrics at the same time will be an important part of the results. We consider the following approaches as interesting to consider in this domain.

Randomization or unique time intervals: The equal rate adjustment intervals (*T*) used by each adaptive stream while in steady-state may be a contributing factor to inaccurate estimations of available bandwidth and thereby oscillating behavior. An alternative to fixed intervals would be to randomize them by using a per-session stochastic parameter (within certain reasonable bounds) or assigning each session a unique value. By doing so the available bandwidth estimation methods may become more accurate.

Back-off periods: Whenever a service is reducing its rate level due to observed congestion it may try to increase again after the same amount of time (*T*). In addition to the previous described randomization/unique approach to this interval, one could also consider introducing a back-off period. This would imply that after a service has reduced its rate level, it enters a back-off period of a certain duration during which, no increase is allowed.

Threshold based behavior: Rather than using the same intervals of potential rate changes all the time, one could introduce a threshold for when it operates more or less aggressive. This threshold could be the mean available rate level for a specific session, or even a smoothed average value for the actual achieved level. This concept is applied with success in more recent TCP versions for the purpose of optimizing performance.

The method chosen for this study is according to the first approach described, i.e., using a random or unique interval between each potential rate change. This would represent a different approach than the default method used in Smooth Streaming from Microsoft [2].

As baseline for the simulations, the default interval *T=2s* has been used. Then as alternatives, both a stochastic distribution and per session fixed unique distribution has been implemented. For the stochastic approach the Uniform distribution was chosen with parameters [1.6, 2.4]s. For the fixed unique approach, the sessions were spread on the following value set [1.6, 1.8, 2.0, 2.2, 2.4]s.

IV. SIMULATION MODEL

As the adaptive streaming solutions of today are highly proprietary, the details concerning their implementation are not disclosed. Due to this, there will always be some degree of uncertainty concerning their internal functions.

The simulation model is based on our earlier work [1] but has been somewhat simplified in order to allow for comparison with experimental results.

A. Assumptions

One of the key functions of an ABR client is the method used for determining whether to go up or down in rate level during times of varying available bandwidth. From studying live traffic it does not seem as if the clients use additional network probing beyond the actual information obtained through download of video segments. Further on, in the likely absence of a per stream traffic shaper at the server side (for scalability and performance reasons), it will give a traffic pattern for each stream which, typically contains a sequence of burst and idle periods. The measured burst period rate is then higher than the actual stream rate level. Also, it is likely that there will be sub-periods within the burst periods where per packet rate is close to the total available bandwidth. As such, the client can probably obtain a rather accurate indication of maximum available bandwidth by just looking at minimum observed inter-arrival time of packets of known size belonging to the same stream.

However, not all streams will have interleaved burst periods so there is a good chance for each stream to overestimate the potential for additional bandwidth. There is a wide range of bandwidth estimation methods and a few of these are described in [22], but again - as the details of the adaptive streaming solutions are not disclosed we will not discuss this part any further. Independent of which, method being used, there will be some degree of uncertainty which, contributes to variable performance. Further on, we assume the following to be true for the ABR sessions to be studied

- No stream coordination at server side
- No involvement from mechanisms in the network between the client and server
- All clients operate independently and do not communicate
- All clients are well behaved in the sense that they follow the same scheme
- At each defined stream rate level there are no variations due to i.e., picture dynamics
- All clients access the same stream on the server side

B. Session Type and Schedule

The ABR sessions used in the simulator are based on profiles observed in commercial services. The quality levels defined are {0, 350, 500, 1000, 1500, 2000, 3000, 4000, 5000} Kbps. All sessions are of the same type. The sessions are initiated by 5 different users and start time scheduling are done according to the stochastic distributed parameter t_a – Uniform [0, 2000]ms. This gives that all sessions start during the first 2 seconds.

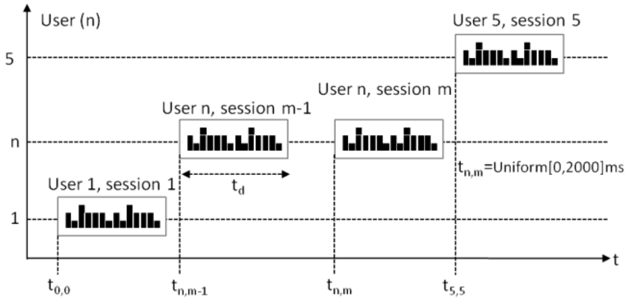

Figure 1. Session scheduling per user

During one simulation run, each user executes a total of 100 sessions sequentially. Time for starting the next session (m) for specific user (n) is noted $t_{n,m}$ (cf. Figure 1). The duration of each session t_d is deterministic and set to 25 minutes.

C. Rate Adaptation Algorithm

The model for rate adaptation per session is based on periodic estimation of available bandwidth $A_s(t)$ and calculation of a smoothed average $SA_s(t)$.

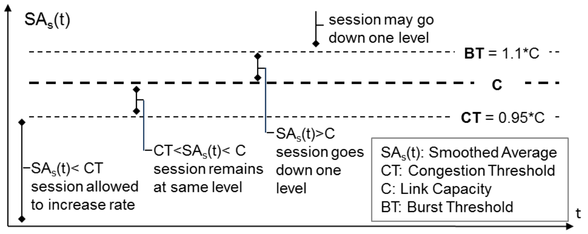
$SA_s(t)$
session may go down one level
BT = 1.1*C
C
CT = 0.95*C
$SA_s(t)$< CT session allowed to increase rate
CT<$SA_s(t)$< C session remains at same level
$SA_s(t)$>C session goes down one level
$SA_s(t)$: Smoothed Average
CT: Congestion Threshold
C: Link Capacity
BT: Burst Threshold
t

Figure 2. Thresholds for smoothed average

This smoothed average (cf. Figure 2) is compared to a congestion threshold (*CT*), the link capacity (*C*) and a burst threshold (*BT*) in order to trigger a rate adjustment.

Whenever the sum of requested rates from sessions is above the burst threshold (*BT*), the next session which, calculates $SA_s(t)$ will be forced down, independent of the value of $SA_s(t)$. This function is implemented in the simulator in order to incorporate the somewhat unpredictable behavior during times of heavy congestion.

The calculation of smoothed average $SA_s(t)$ is based on [5], and is expressed in (1). The parameter δ gives the weighting of the estimated available bandwidth for the two periods included in the calculation.

$$SA_s(t) = \delta\, A_s(t_{i-1}) + (1-\delta)A_s(t_i) \qquad (1)$$

The available bandwidth estimation function used in the simulations is based on the assumption that sessions running at high rates are able to make more accurate estimations than those running at lower rates. An abstraction of the function itself is made by a number of n bandwidth samples $C_{i,j}$ (cf. Figure 3)

A specific session is then given access to a number of these samples according to its current rate level, and then it will use this as basis for its estimation. A high rate gives a high number of samples available, and then, also, a higher degree of accuracy.

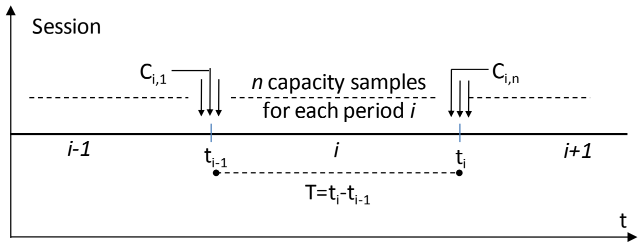

Figure 3. Capacity samples per period

The number of samples $x_{s,i}$ available to a specific session *s* for period *i* is given by its ratio between current rate $R_s(t_i)$ and max rate $R_s max$, multiplied by *n* as per (2).

$$x_{s,i} = n\, {}^{R_s(t_i)}\!/_{R_s max} \qquad (2)$$

In the simulations, the value of n was set to 20 and $R_s max$ was according to the session definition 5000Kbps. The available bandwidth estimated $A_s(t)$ for period *i* is then given by the following (3).

$$A_s(t_i) = \sum_{l=1}^{x_i} {}^{C_{i,l}}\!/_{x_{s,i}} \qquad (3)$$

By combination with the expression for $SA_s(t)$ it gives the following expression (4).

$$SA_s(t) = \delta \sum_{l=1}^{x_{i-1}} {}^{C_{i,l}}\!/_{x_{s,i-1}} + (1-\delta)\sum_{l=1}^{x_i} {}^{C_{i,l}}\!/_{x_{s,i}} \qquad (4)$$

The value of δ was set to 0.8 as per [5], thus giving most weight to the available bandwidth estimation from the previous period.

D. Simulation Tool

The simulator was built using the process oriented Simula [23] programming language and the Discrete Event Modeling On Simula (DEMOS) context class [24].

This programming language is considered as one of the first object oriented programming languages, and remains a strong tool for performing simulations.

V. Simulation results

The simulation results are presented for different capacity levels on the access link. The chosen capacities are 10, 12.5, 15, 17.5 and 20Mbps. At all these capacity levels there would be congestion as the sum of the maximum quality level requested for the 5 competing sessions is 25Mbps. The simulations were also run for capacity levels between those given above, but for the sake of clarity these details are left out as they did not change the conclusions.

Simulation session results are sorted and then grouped according to the studied metrics, giving a clear view on performance ranging from the worst to the best performer.

The characterization is done by looking at the distributional properties location (mean), spread (mid 50% values) and high/low 25% results. For this purpose the box and whisker plots are used as they give a compact view of all these properties.

A. Proportional Fairness

As defined, proportional fairness is calculated by the achieved session average rate per user, divided by session max – and then a comparison of these values are done for the competing sessions/users. The results from the simulations give 100 independent samples for this metric.

Improvements in proportional fairness are then recognized as reduced difference between the worst and best performing sessions. The results are presented in Figure 4, Figure 5 and Figure 6 showing both the mean values and the spread of the metric sample distributions.

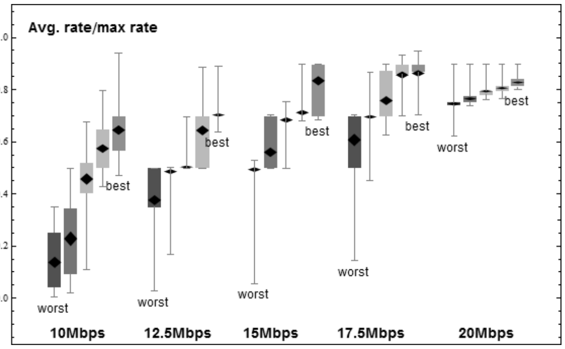

Figure 4. Proportional Fairness, Equal T, Simulations

The results shown in Figure 4 illustrates that there is a significant challenge in terms of proportional fairness when using the default equal T approach for all access capacity levels except for at the highest level (20Mbps).

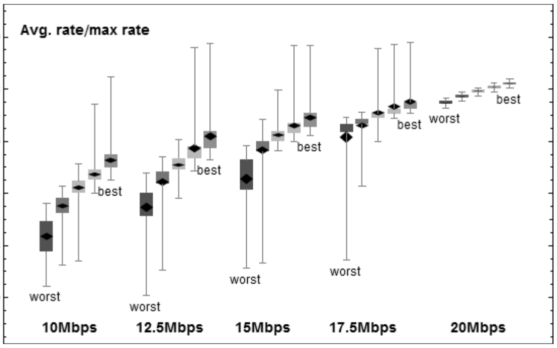

Figure 5. Proportional Fairness, Unique T, Simulations

By careful study of the results shown in Figure 5 for the unique T approach one can see that the difference between the worst and best performing sessions are reduced, and thereby an improved proportional fairness.

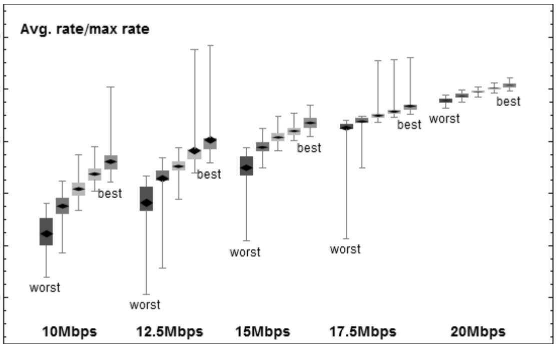

Figure 6. Proportional Fairness, Random T, Simulations

The same effect as for the unique T approach is also visible for the random T approach as shown in Figure 6. For both approaches it is also worth noticing the reduced spread of observations as indicated by the mid 50% values.

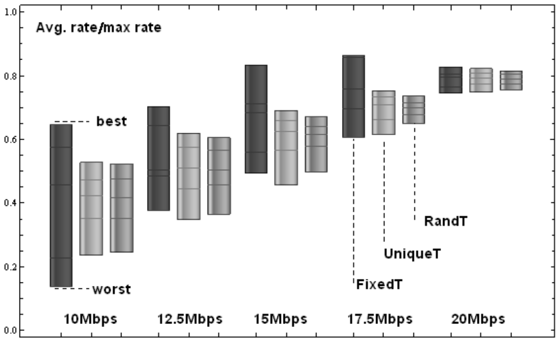

Figure 7. Summary Proportional Fairness, Simulations

In the summary view of the simulations results as shown in Figure 7 the effect of both random T and unique T methods are quite clear. The differences between the competing sessions become smaller, thus we can state that the simulations give reason to believe that the methods studied give improvements in terms of proportional fairness.

B. *Perceived Fairness*

As follows by our definition of perceived fairness a small difference between sessions in terms of number of rate reductions per minute is good. The rationale behind this would be an assumption of that different users have insight into the performance of other sessions. In addition, the absolute value is of course also important. A low metric value is good.

The results shown in Figure 8 give a clear indication on that the simulator model is quite aggressive in terms of how often it allows each stream change its quality level. The level of 15 reductions / minute is likely to represents the model maximum. This follows by T intervals of 2 sec, and our presentation of reductions / minute only.

The results for perceived fairness using the equal T approach are quite poor in the sense that the absolute values are at maximum level for the three mid capacity levels. However, it should be noted that the simulator model contains some simplifications and assumptions which may not be accurate enough in this domain.

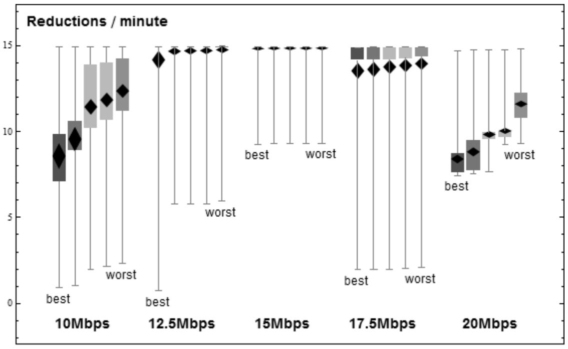

Figure 8. Perceived Fairness, Equal T, Simulations

The results shown Figure 9 for the unique T approach illustrates that the reductions per minute are reduced, but at the same time it introduces a stronger difference between sessions.

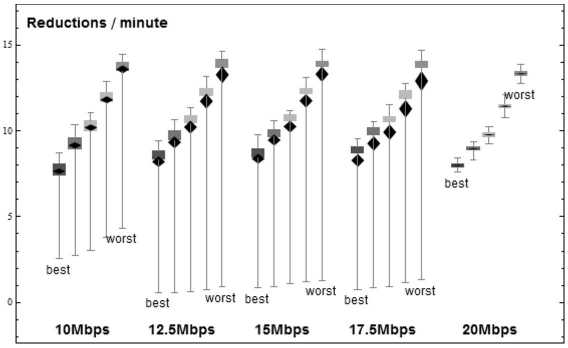

Figure 9. Perceived Fairness, Unique T, Simulations

The same effect as for the unique T approach is also visible for the random T approach as shown Figure 10. Except for the higher spread at 20Mbpss capacity levels, the results are quite similar.

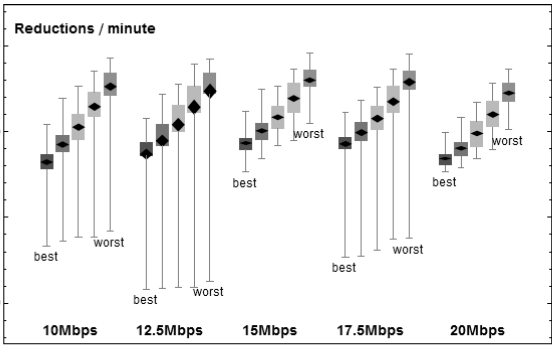

Figure 10. Perceived Fairness, Random T, Simulations

The summary view of perceived fairness is presented in Figure 11. It illustrates both the actual values for the best/worst performers and the difference between them. Results are presented for the default equal T, unique T and random T methods for all access capacity levels. These results alone do not give reason to believe that the investigated method (unique T and random T) give an improved perceived fairness.

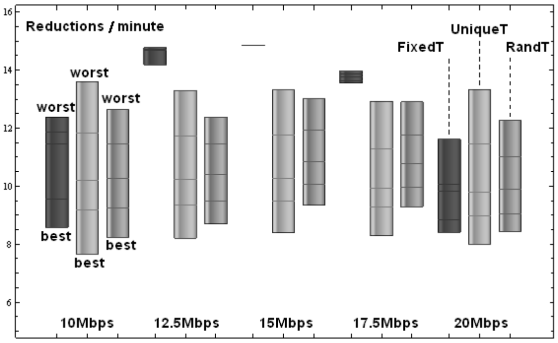

Figure 11. Summary Perceived Fairness, Simulations

VI. MEASUREMENTS

In order to perform the required measurements a testbed was established in a controlled environment including all required componenst [10].

As illustrated in Figure 12 the 5 clients are located behind a shared access with a certain capacity towards a Microsoft Smooth Streaming service. This scenario matches the one which was built into the simulator as described in section IV.

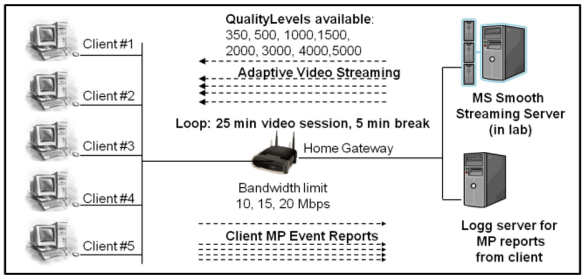

Figure 12. Measurement setup

The clients used were separate PC's with identical HW and SW and set to access the same adaptive HTTP video stream from the lab server. Controlled by scripts on each PC the clients were 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 [25] from each client (Monitor Plane event reports) was used in order to record interesting events on a per session basis and allow for effective post processing.

The measurements results for proportional fairness and perceived fairness are given in the following sections using the same presentation form as for the simulations.

A. *Proportional Fairness*

The results shown in Figure 13 illustrate that there is a problem with regards to proportional fairness when using the default equal T approach for all access capacity levels. The problem is smallest at the highest level (20Mbps), which matches the earlier presented simulation results (cf. Figure 4).

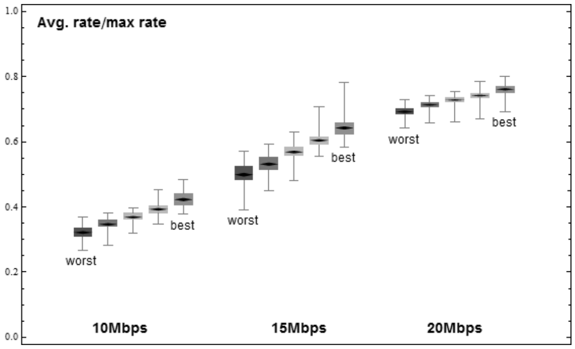

Figure 13. Proportional Fairness, Equal T, Measurements

By studying the results shown in Figure 14 for the unique T approach, a noticeable reduced difference between the worst and best performing sessions are seen. This again, is in line with the corresponding simulation results (cf. Figure 5) indicating improved proportional fairness.

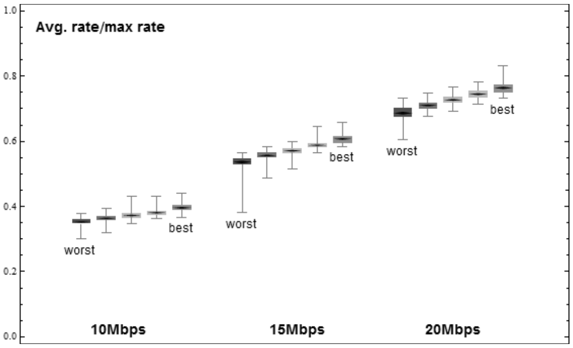

Figure 14. Proportional fairness, Unique T, Measurements

A similar positive effect as for the unique T approach is also visible for the random T approach (cf. Figure 15).

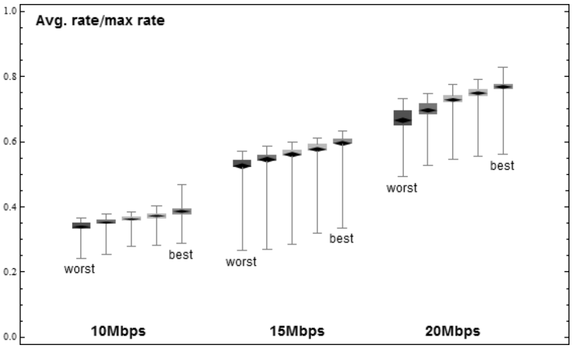

Figure 15. Proportional Fairness, Random T, Measurements

In the summary view of the measurements results as shown in Figure 16 the positive effect of both random T and unique T methods are quite clear, except for at the highest access capacity level (20Mbps). These findings are much in line with the finding from the simulations (cf. Figure 7).

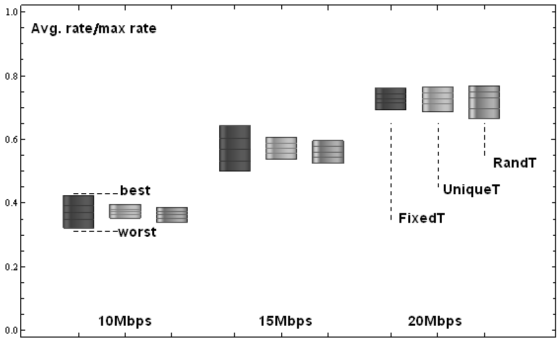

Figure 16. Summary Proportional Fairness, Measurements

It should be noted that measurements were not performed for all the capacity levels which were used in the simulations. The main reason for this is the amount of time required for performing measurements versus time required for simulations.

B. Perceived Fairness

The first thing which is noticed when looking at the measurements result for perceived fairness in Figure 17 is that the levels observed are much lower than those collected during simulations (cf. Figure 8). Thereafter, one can see that there is a clear difference between the best and worst performing sessions but the absolute values are rather low.

Therefore, based on these findings we can only state that there is a pure theoretical challenge with perceived fairness. Whether actual users will feel discriminated or get a poor user experience due to quality fluctuations at these levels is not evident.

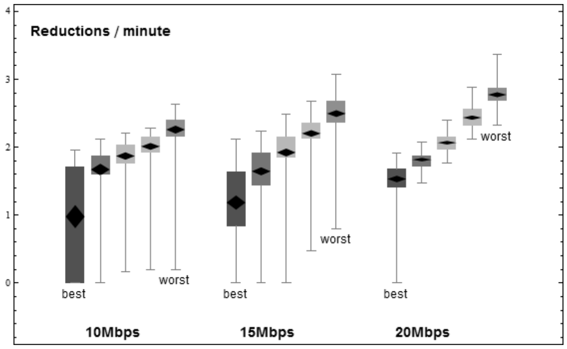

Figure 17. Perceived Fairness, Equal T, Measurements

The results shown Figure 18 for the unique T approach illustrates that the spread in the observations are reduced (mid 50% observations), but the mean value levels remain in the same regions as for the default equal T approach.

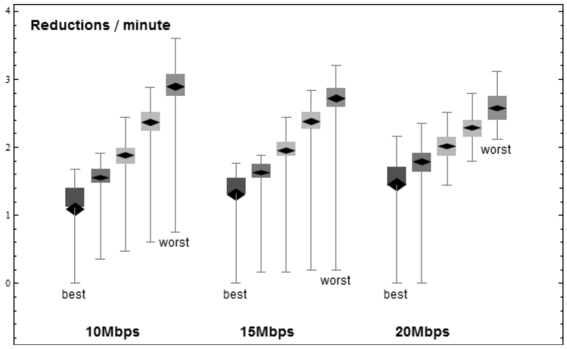

Figure 18. Perceived Fairness, Unique T, Measurements

For the random T approach as illustrated in Figure 19 we see an increase in spread for the observations at the two lowest access capacity levels, making the results in this regard almost similar to the default equal T approach. The exception is the results for 20Mbps access where a quite clear positive effect is seen with regards to difference between the worst and best performing session.

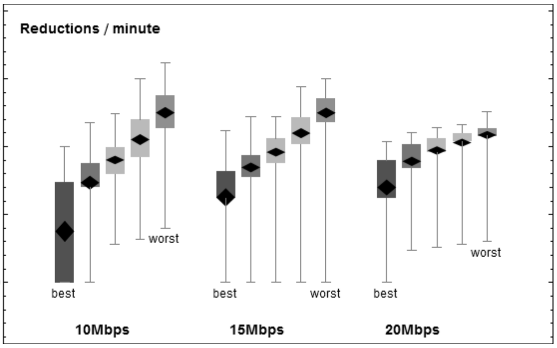

Figure 19. Perceived Fairness, Random T, Measurements

The summary view of perceived fairness based on measurements is presented in Figure 20. As can be seen, the results do not give reason to state an improvement in terms of perceived fairness when implementing either the unique T or random T methods.

These findings are in line with the simulation results, although there is a major difference in the absolute levels.

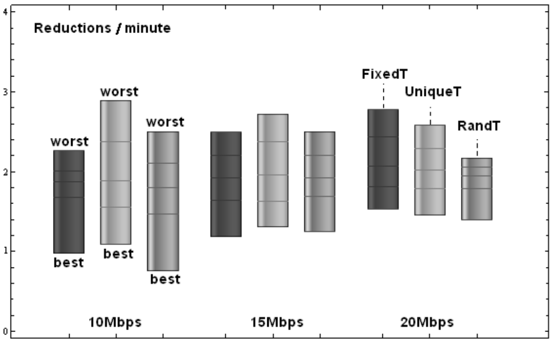

Figure 20. Summary Perceived Fairness, Measurements

VII. COMPARING SIMULATIONS AND MEASUREMENTS

The results from simulations and measurements differ in absolute values for both proportional fairness and perceived fairness. Keeping in mind that any simulation is based on a model and not the real system itself, this does not come as surprise. However, the important thing is to highlight the effect of introducing the suggested methods (random T, unique T) and see if there are similarities in this regard in both the simulation and measurement domains.

Looking at the combined results for proportional fairness given in Figure 21 we see that a similar effect is present in both domains. There is a clear positive effect of introducing either the random T or unique T method.

Both the simulation results and measurements results show a very strong positive effect for most access capacity levels, except for at the highest level (20Mbps) where the effect is close to neutral.

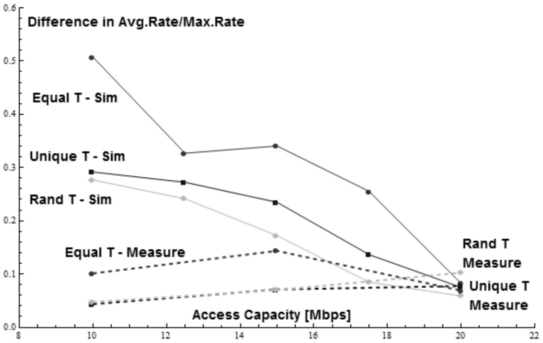

Figure 21. Proportional Fairness, Measurements and Simulations

For the perceived fairness results as shown in Figure 22 the simulation part indicates a strong improvement for the suggested methods. However, as the absolute values are so high (close to assumed maximum) the credibility of these results is weakened. The rate adaptation algorithm implemented in the simulator is probably too aggressive compared to the real life implementations.

The measurement results for perceived fairness are neutral viewed alone, but when combined with the proportional fairness results one can say it is positive that improvements in the pure QoS domain does not come at the expense of degraded performance in the QoE domain.

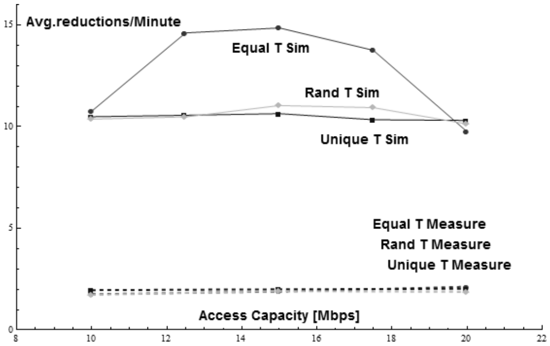

Figure 22. Perceived Fairness, Measurements and Simulations

In summary, the simulations together with the measurements gives a strong indication of that the suggested methods have a potential real life value in terms of improving proportional fairness.

The differences between using a random T value or a per session unique T value does not give basis for saying which is better. However, from an implementation point of view the random approach clearly has its challenges as the video content requires encoding according to these intervals. In light of this, the approach of using per session unique T values is the preferred one.

VIII. ANALYSIS

The somewhat intuitive explanation to why changes could be expected when introducing either a random T or unique T rate adjustment interval is that some of the negative effects of an equal adjustment interval as illustrated in Figure 23 are reduced. In the case of equal periods, each session would get the same periodic view on the link utilization, always missing or including some other traffic. This gives a certain degree of error in the available bandwidth estimation functions.

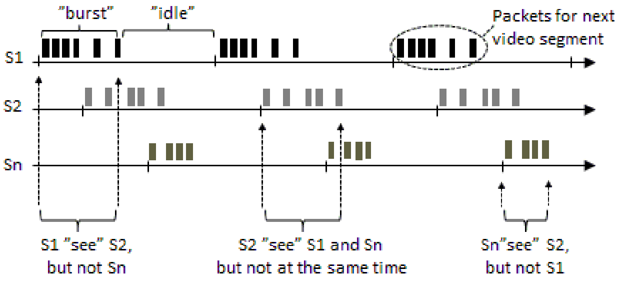
"burst"
"idle"
Packets for next video segment
S1
S2
Sn
S1 "see" S2, but not Sn
S2 "see" S1 and Sn but not at the same time
Sn"see" S2, but not S1

Figure 23. Problem with equal estimation periods

Each session estimates available bandwidth only during its burst periods. Although not explicitly stated in system documentation [2] this has been verified in the measurement testbed [10]. As part of this work the absence of any active network probing was verified. Therefore, whatever notion of available bandwidth each client uses as basis for its rate adaptation it must be based on information collected during the periods where it receives video segments (burst periods). This means that in order to get an accurate estimation it is beneficial for each client to have overlapping burst periods with as many other sessions as possible.

A. *Burst Period Duration*

The duration of the burst period for a specific session depends on both its current rate level and the rate adjustment interval. The dependency of the rate level follows from the obvious relation to data volume to be transferred per time unit for a specific rate level, while the dependency of rate adjustment interval follows from the requirement to maintain the same average amount of data received over time.

At the beginning of each interval the client requests the next video fragment for a specific rate level, with duration equal to its rate adjustment interval. This is illustrated in Figure 24 where two sessions running at the same rate level, but with different rate adjustment intervals have different burst period durations.

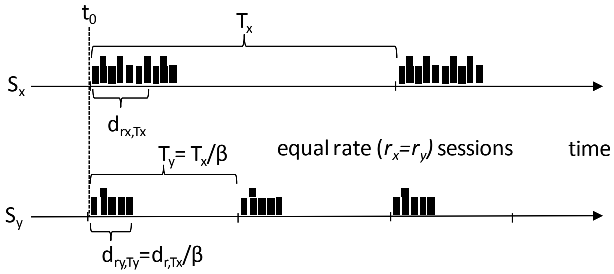
t_0
T_x
S_x
$d_{rx,Tx}$
$T_y= T_x/\beta$
equal rate (r_x=r_y) sessions
time
S_y
$d_{ry,Ty}$=$d_{r,Tx}/\beta$

Figure 24. Equal rate sessions with different burst periods

Any two sessions (S_x, S_y) running at the same rate level, will have a relation between their burst period durations expressed by the parameter β. This parameter is given by the following expression (5).

$$\beta = {d_{rx,Tx}}/{d_{ry,Ty}} = {T_x}/{T_y}, \qquad for\ r_x = r_y \tag{5}$$

Using this relationship, we can express (6) the burst period duration $d_{ri,Ti}$ for any session S_i as a function of its rate adjustment interval T_i and a reference burst period duration $d_{ri,T}$.

$$d_{ri,Ti} = d_{ri,T}({T_i}/{T}) \tag{6}$$

The values for $d_{ri,T}$ can presumably be calculated based on information about the codec used for the specific media stream inside each sessions, together with assumption on per session server side capacity. Alternatively one could make measurements on a specific system and establish a $d_{ri,T}$ matrix for all valid values of r_i and the reference T value.

However, if we assume that the server side capacity is not a limitation, and that it will always try to burst with a certain bitrate C_{burst} we can also express the burst period duration $d_{ri,Ti}$ as follows (7).

$$d_{ri,Ti} = \left({r_i\,T}/{C_{burst}}\right)\left({T_i}/{T}\right) = \left({r_iT_i}/{C_{burst}}\right) \tag{7}$$

The maximum value for C_{burst} is natural to think of as the access capacity for the user group / home network, as this is normally the end-to-end bottleneck. However, it is likely that the actual C_{burst} is related to the maximum rate for the specific service.

B. *Probability for Burst Period Overlap*

For T_i values according to a uniform distribution, the probability $P_{i,r,t}$ for a session i at rate level r to be in its burst period at time t will be according to the following expression (8).

$$P_{i,r,t} = {d_{r,Ti}}/{T_i} = \frac{d_{r,T}\left({T_i}/{T}\right)}{T_i} = {d_{r,T}}/{T} \tag{8}$$

From this, we see that all sessions at a specific rate level has the same probability of being in its burst period at time t. We can then express the probability that all n sessions are in their burst period at time t as follows (9).

$$P_{all\ burst,t} = \left({d_{r_1,T}}/{T}\right)^{c_1} \quad \left({d_{r_m,T}}/{T}\right)^{c_m} \tag{9}$$

The parameter c_m represents the number of sessions at rate level r_m and the sum of all c_m values equals n. From this we see that the probability of any session to see all other sessions during its burst period depends on the session rate level mix, and this probability increases when more sessions are running at high rate levels.

Further on, we recognize that the probability for that a session i has an overlap with each of the other sessions sometimes during its burst period T_i is the integral of $P_{all\ burst,t}$ over the period [0, T_i] which, is easily expressed as the constant $P_{all\ burst,t}$ multiplied by T_i.

We then let a specific session mix be described by the vector $R_{mix}=\{r_1,\ldots,r_n\}$, whereas r_i represents the rate level for session i. Also, for a specific session i let A_i be the group of sessions which, has overlapping burst periods with session i at a specific time t_0, and B_i be the group of sessions for which, it did not have an overlap. In the situation where all sessions have the same rate adjustment interval duration T_i, the probability of that session i has an overlapping burst period with any of the sessions in group B_i at time t_0+T_i is zero. This leads to that while R_{mix} remains unchanged, the view a specific session has of the total traffic will not change. The system state for session i in terms of burst period overlap with other sessions is independent of the state at t_0 and also t in general.

In the case where T_i is not equal for all sessions, but instead are chosen according to some stochastic distribution – the group of sessions which, overlap the burst period of session i at t_0+T_i is not independent of the state at t_0. If we let C_i denote the sub-group of sessions from B_i which, has overlapping burst periods with session i at time t_0+T_i, it can be shown that there is a deterministic relationship between A_i, B_i and C_i.

If we then remember the assumed use of a smoothed average function we see the benefit of this potential additional burst period overlaps in subsequent periods.

C. Dynamics in Burst Period Overlap

When the starting times for each session and their respective rate adjustment intervals (T_i) are considered stochastic processes, the sessions will combine in time in different ways. In order to define the deterministic relationship between overlapping burst periods during subsequent intervals, we need to analyze scenarios where sessions with different rate levels and different rate adjustment interval are combined.

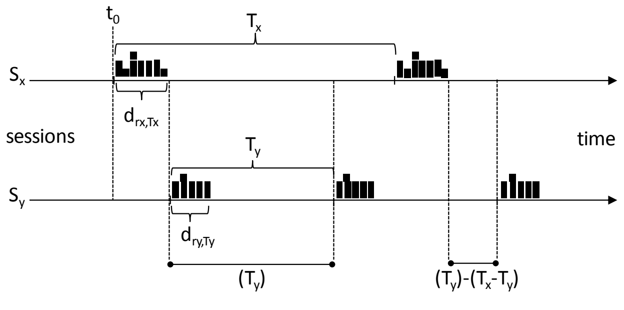

Figure 25. Session S_y staring after S_x (T_y<T_x)

The first scenario (a) to be studied is the one where two sessions (S_x, S_y) with different T_i values (T_x, T_y) are active at the same time. We assume $T_x > T_y$ and that S_y starts immediately after the burst period of S_x finishes as illustrated in Figure 25.

For the two sessions (S_x, S_y) there will be shift in phase between them as a function of time which, makes them have a full or partial burst period overlap at some time. The question is then how many rounds it will take for S_x to see S_y and vice versa. It can be shown that we can express the number of rounds for S_x before it has an overlapping burst period with S_y as follows (10).

$$\begin{gathered}\mathrm{N}_{\mathrm{a,x\rightarrow y}} = 1 + \left\lceil \frac{T_y}{T_x - T_y} \right\rceil \\ \text{when } \tfrac{T_x}{2} < T_y < (T_x - d_{rx,Tx} - d_{ry,Ty}) \\ \mathrm{N}_{\mathrm{a,x\rightarrow y}} = 2 \\ \text{when } (T_x - d_{rx,Tx} - d_{ry,Ty}) < T_y < T_x\end{gathered} \qquad (10)$$

In the same way, we can express the number of rounds for S_y before the same overlap of burst period with S_x takes place (11).

$$\begin{gathered}\mathrm{N}_{\mathrm{a,y\rightarrow x}} = 1 + \left\lceil \frac{T_x}{T_x - T_y} \right\rceil \\ \text{when } \tfrac{T_x}{2} < T_y < \left(T_x - d_{rx,Tx} - d_{ry,Ty}\right) \\ \mathrm{N}_{\mathrm{a,y\rightarrow x}} = 2 \\ \text{when } \left(T_x - d_{rx,Tx} - d_{ry,Ty}\right) < T_y < T_x\end{gathered} \qquad (11)$$

The next scenario (b) to be studied is where the sessions (S_x, S_y) are running with different T_i values (T_x, T_y) but now S_y finishes its burst period before S_x (cf. Figure 26).

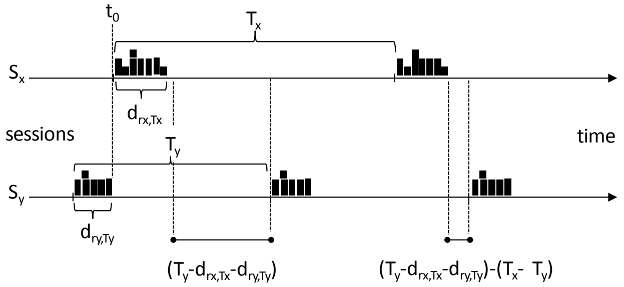

Figure 26. Session S_x staring after S_y (T_y<T_x)

The number of rounds it takes for S_x to see S_y is expressed as follows (12).

$$\begin{gathered}\mathrm{N}_{\mathrm{b,x\rightarrow y}} = 1 + \left\lceil \frac{T_y - dx - dy}{T_x - T_y} \right\rceil \\ \text{when } \tfrac{T_x}{2} < T_y < \left(T_x - d_{r,Ty}\right) \\ \mathrm{N}_{\mathrm{b,x\rightarrow y}} = 2 \\ \text{when } (T_x - d_{r,Ty}) < T_y < T_x\end{gathered} \qquad (12)$$

The number of rounds it takes for S_y to see S_x is expressed as follows (13).

$$\mathrm{N}_{\mathrm{b,y\to x}} = 1 + \left\lceil \frac{T_x - dx - dy}{T_x - T_y} \right\rceil$$
$$\text{when } \tfrac{T_x}{2} < T_y < (T_x - d_{r,Ty}) \qquad (13)$$
$$\mathrm{N}_{\mathrm{b,y\to x}} = 2$$
$$\text{when } (T_x - d_{r,Ty}) < T_y < T_x$$

It should be noted that for both scenarios there is a special case where $\mathrm{N}_{\mathrm{a,y\text{-}x}}/\mathrm{N}_{\mathrm{b,y\text{-}x}}$ and $\mathrm{N}_{\mathrm{a,x\text{-}y}}/\mathrm{N}_{\mathrm{b,x\text{-}y}}$ are always 2, i.e., two sessions which, did not have overlapping burst periods at t_0 is guaranteed to have overlapped during the next period for S_x and S_y. For a smoothed average function operating over two periods this is desirable, i.e., whatever it does not see in the first period it is guaranteed to see in the next.

D. Optimization Problem

The expressions for $N_{y\text{-}x}$ and $N_{x\text{-}y}$ contain many variables. These variables are the rate adjustment intervals T_i and the burst period durations $d_{ri,Ti}$ for all sessions. The latter are calculated based on the session rates r_x and r_y and C_{burst} as defined in Section V. These expressions can be used as input to a constrained optimization problem and analyzed as such in order to find maximum and minimum values.

As the starting point for this optimization problem we can focus on the worst case scenario, that would be the number of rounds for S_y before it has an overlap with S_x ($\mathrm{N}_{\mathrm{a,y\text{-}x}}/\mathrm{N}_{\mathrm{b,y\text{-}x}}$), which, will always be higher than the number of rounds for S_x before this has an overlap with S_y.

We also see that $\mathrm{N}_{\mathrm{a,y\text{-}x}}$ will always be greater than $N_{b,y\text{-}x}$ since $T_x > T_y$. This gives us only one expression to analyze for the worst case scenario as follows (14).

Maximize: $N_{a,y\to x}$

where

$$N_{a,y\to x} = \begin{cases} 1 + \left\lceil \dfrac{T_x}{T_x - T_y} \right\rceil, if\ T_y < \left(T_x - d_{rx,Tx} - d_{ry,Ty}\right) \\ 2, if\ \left(T_x - d_{rx,Tx} - d_{ry,Ty}\right) < T_y < T_x \end{cases}$$

subject to: (14)

$$1.6 < T_y, T_x < 2.4 \text{ and } T_x/2 < T_y$$
$$r_x, r_y \in \{350, 500, 1000, 1500, 2000, 3000, 4000, 5000\}$$
$$d_{r_x,T_x} = {}^{r_xT_x}/_{C_{burst}}$$
$$d_{r_y,T_y} = {}^{r_yT_y}/_{C_{burst}}$$

The above maximization can then be done for different values of C_{burst}. In the simulations and measurements the access capacities used were between 10 and 20Mbps and the maximum session rate was 5Mbps. Based on measurements of real traffic we can see that the C_{burst} is lower than the actual access speed and therefore values of respectively 5Mbps, 7.5Mbps and 10Mbps were used for C_{burst}.

For the two different alternatives of choosing values for T_i used in the simulations and measurements, both the random T and unique T approaches are possible to work with in the optimization context. However, as the unique T approach will be a special case (subset) of the random T, we only present results for the random T approach.

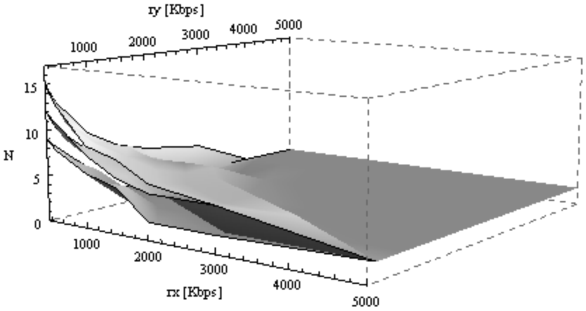

Figure 27. Maximum $\mathrm{N}_{\mathrm{a,y\text{-}x}}$ for different burst bitrates

The result from solving the optimization problem is shown in Figure 27. The three different burst bitrates (C_{burst}) give surfaces which, are plotted, whereas the highest capacity gives the highest values for $N_{a,y\text{-}x}$.

We see that in many cases we get an overlap already in the second round, and thereby we improve the basis for the available bandwidth estimation algorithm. This analysis then strengthens the findings in both the simulations and measurements.

In order to improve the available bandwidth estimations further one may consider the well known PASTA principle [26] from queuing theory which, states that a Poisson based Arrival process See Time Averages. This implies that the bandwidth probing should take place not only during the burst periods, but as a process taking samples throughout the whole rate adjustment period. However, as this implies some degree of active probing it would potentially have some other undesirable effects.

IX. Conclusions and Future Work

The results show that there is a significant potential of improving proportional fairness as defined while maintaining perceived fairness for adaptive streams of the category studied. The positive effect of the suggested enhancement to the rate adaptation scheme, i.e., using a random or unique duration of rate adjustment intervals rather than the default equal value is supported by simulation results, measurements and also rationalized by the theoretical analysis. The findings differ to some extent from those in our previous work [1], but at the same time we now have a more refined and accurate view of the methods studied. The added value of results from measurements has been significant.

The results also illustrate that when studying the performance of adaptive streaming solutions, it is not enough to only focus on the network centric QoS domain. A change in this domain does not necessary lead to a corresponding change in the QoE domain, and vice versa.

As future work in this field it is planned to further study objective and no-reference based QoE metrics which, is possible to correlate over to the QoS and network domain. It is also planned to study various available bandwidth algorithms with regard to their real-time capabilities and thereby suitability for adaptive video streaming.

X. 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.

The work has also been actively supported by TV2, the leading commercial TV broadcaster in Norway. TV2 is among the pioneers in providing a full commercial TV offering over the Internet based on ABR technology.

References

[1] B. Villa and P. Heegaard, “Improving perceived fairness and QoE for adaptive video streams,” *ICNS 2012*, March 2012. In Proceedings of ICNS 2012: The Eighth International Conference on Networking and Services, Thinkmind 2012, ISBN 978-1-61208-186-1, March 2012, pp. 149-158.

[2] A. Zambelli, “IIS Smooth Streaming Technical Overview,” Tech. Rep., March 2009.

[3] ISO, *Dynamic adaptive streaming over HTTP (DASH). ISO/IEC FCD 23001-6*, International Organization for Standardization Std. ISO/IEC FCD 23001-6, 2011.

[4] E. Areizaga, L. Perez, C. Verikoukis, N. Zorba, E. Jacob, and P. Odling, “A road to media-aware user-dependent self-adaptive networks,” in *Proc. IEEE Int. Symp. BMSB ’09*, 2009, pp. 1–6.

[5] S. Akhshabi, A. C. Begen, and C. Dovrolis, “An experimental evaluation of rate-adaptation algorithms in adaptive streaming over http,” in *ACM Multimedia Systems (MMSys)*, 2011.

[6] L. De Cicco and S. Mascolo, “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*, ser. USAB’10. Berlin, Heidelberg: Springer-Verlag, 2010, pp. 447–464.

[7] R. Kuschnig, I. Kofler, and H. Hellwagner, “An evaluation of tcp-based rate-control algorithms for adaptive internet streaming of h.264/svc,” in *MMSys ’10*. New York, NY, USA: ACM, 2010, pp. 157–168.

[8] B. J. Villa and P. E. Heegaard, “Monitoring and control of QoE in media streams using the click software router,” in *NIK2010, Norway. ISBN 978-82-519-2702-4.*, vol. 1, November 2010, pp. 24–33.

[9] B. Villa and P. Heegaard, “Towards knowledge-driven QoE optimization in home gateways,” *ICNS 2011*, May 2011, Thinkmind, ISBN 978-1-61208-133-5, pp. 252-256.

[10] B. J. Villa, P. E. Heegaard, and A. Instefjord, “Improving fairness for adaptive http video streaming,” in *EUNICE* 2012, Springer 2012, ISBN 978-3-642-32807-7, pp. 183–193.

[11] S. Bhatti, M. Bateman, and D. Miras, “Revisiting inter-flow fairness,” in *Broadband Communications, Networks and Systems, 2008. BROADNETS 2008. 5th International Conference on*, sept. 2008, pp. 585 –592.

[12] S. Floyd, “Metrics for the Evaluation of Congestion Control Mechanisms,” RFC 5166 (Informational), Internet Engineering Task Force, Mar. 2008.

[13] R. Jain, D. Chiu, and W. Hawe, “A quantitative measure of fairness and discrimination for resource allocation in shared systems,” DEC, Tech. Rep., 1984.

[14] D. Mitra and J. B. Seery, “Dynamic adaptive windows for high speed data networks with multiple paths and propagation delays,” *Computer Networks and ISDN Systems*, vol. 25, no. 6, pp. 663 – 679, 1993, high Speed Networks.

[15] Y. Zhang, S.-R. Kang, and D. Loguinov, “Delayed stability and performance of distributed congestion control,” in *Proceedings of the 2004 conference on Applications, technologies, architectures, and protocols for computer communications*, ser. SIGCOMM ’04. New York, NY, USA: ACM, 2004, pp. 307–318.

[16] Z. Cao and E. Zegura, “Utility max-min: an application-oriented bandwidth allocation scheme,” in *INFOCOM ’99*, vol. 2, mar 1999, pp. 793 –801 vol.2.

[17] F. Kelly, “Charging and rate control for elastic traffic,” *European Transactions on Telecommunications*, 1997.

[18] S. Kunniyur and R. Srikant, “End-to-end congestion control schemes: utility functions, random losses and ecn marks,” *IEEE/ACM Trans. Netw.*, vol. 11, pp. 689–702, October 2003.

[19] A. Jdidi and T. Chahed, “Flow-level performance of proportional fairness with hierarchical modulation in ofdma-based networks,” *Comput. Netw.*, vol. 55, pp. 1784–1793, June 2011.

[20] H. Levy, B. Avi-Itzhak, and D. Raz, “Network performance engineering,” D. D. Kouvatsos, Ed. Berlin, Heidelberg: Springer-Verlag, 2011, ch. Principles of fairness quantification in queueing systems, pp. 284–300.

[21] W. Sandmann, “Quantitative fairness for assessing perceived service quality in queues,” *Journal Operational Research*, pp. 1–34, April 2011.

[22] R. Prasad, C. Dovrolis, M. Murray, and K. Claffy, “Bandwidth estimation: metrics, measurement techniques, and tools,” *Network, IEEE*, vol. 17, no. 6, pp. 27 – 35, nov.-dec. 2003.

[23] R. J. Pooley, *An Introduction to Programming in Simula*. Blackwell Scientific Publications. ISBN: 0632014229, 1987.

[24] G. Birtwistle, *Demos - A system for Discrete Event Modelling on Simula*, G. Birtwistle, Ed. School of Computer Science, University of Sheffeld,, July 1997.

[25] B. J. Villa and P. E. Heegaard, “A monitor plane component for adaptive video streaming,” in *NIK2011, Norway. ISSN 1892-0713*, vol. 1, November 2011, pp. 145–154.

[26] R. W. Wolff, “Poisson Arrivals See Time Averages,” *Operations Research*, vol. 30, no. 2, pp. 223–231, 1982.